

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky

BAKALÁŘSKÁ PRÁCE

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a biomedicínského inženýrství

**Zpracování řečových signálů měřicími mikrofony pomocí NI 9234
v LabVIEW**

**Speech Signal Processing using Measuring Microphones NI 9234
and LabVIEW**

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a biomedicínského inženýrství

Zadání bakalářské práce

Student: **Michal Skácel**
Studijní program: **B2649 Elektrotechnika**
Studijní obor: **2612R041 Řídicí a informační systémy**
Téma: **Zpracování řečových signálů měřicími mikrofony pomocí NI 9234
v LabVIEW
Speech Signal Processing using Measuring Microphones NI 9234
and LabVIEW**
Jazyk vypracování: **čeština**

Zásady pro vypracování:

Bakalářská práce je zaměřena na zpracování řečových signálů měřicími mikrofony (G.R.A.S. 40PP) pomocí NI 9234 v LabVIEW. Cílem absolventské práce je návrh a softwarová realizace aplikace pro záznam a analýzu zvukových signálů pomocí dvou mikrofónů (primární a referenční) pro účely adaptivního zpracování ve vývojovém prostředí LabVIEW. Pomocí navržené aplikace bude vytvořena databáze reálných záznamů v různých prostředích pro účely adaptivního zpracování.

Body zadání:

1. Literární rešerše problematiky zpracování řečových signálů v časové a frekvenční oblasti.
2. Popis měřicích mikrofónů G.R.A.S. 40PP a karty NI 9234 z pohledu záznamu řečových signálů pro následné adaptivní zpracování.
3. Návrh a realizace aplikace pro záznam a analýzu zvukových signálů pomocí dvou mikrofónů (primární a referenční) pro účely adaptivního zpracování ve vývojovém prostředí LabVIEW. Aplikace by měla umožňovat analýzu v časové a frekvenční oblasti, dále pak ukládání naměřených dat.
4. Tvorba databáze řečových signálů a rušení (např. ulice, dopravní prostředky, výrobní haly, přednáškové místnosti, sportovní utkání, apod.) pro testování adaptivních metod zpracování řečových signálů.
5. Ověření funkčnosti vytvořené aplikace z pohledu adaptivního zpracování pomocí NI LabVIEW Adaptive Filter Toolkit.
6. Diskuze dosažených výsledků.

Seznam doporučené odborné literatury:

- [1] WITTASSEK, Tomáš. Virtuální instrumentace I: učební text [CD-ROM]. 1. vyd. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, 2014 [cit. 2016-10-18]. ISBN 978-80-248-3395-8.
- [2] SOVKA, Pavel a Petr POLLÁK. Vybrané metody číslicového zpracování signálů. Vyd. 2. přeprac. Praha: Vydavatelství ČVUT, 2003. ISBN 80-01-02821-6.
- [3] UHLÍŘ, Jan. Technologie hlasových komunikací. Vyd. 1. Praha: Nakladatelství ČVUT, 2007. ISBN 978-80-01-03888-8.
- [4] KEHTARNAVAZ, Nasser and Namjin KIM. *Digital signal processing system-level design using LabVIEW*. Burlington(USA): Newnes (Elsevir), 2011. ISBN 978-0-7506-7914-5.
- [5] FARHANG-BOROUJENY, B. Adaptive filters: theory and applications. 2nd ed. Chichester: Wiley, 2013. ISBN 978-1-119-97954-8.
- [6] MARTINEK, Radek, et al. A robust approach for acoustic noise suppression in speech using ANFIS.

Journal of Electrical Engineering, vol.66, iss.6, 2015, p. 301-310. DOI: 10.2478/jee-2015-0050, Print ISSN 1335-3632, On-line ISSN 1339-309X.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Radek Martinek, Ph.D.**

Datum zadání: 01.09.2016

Datum odevzdání: 28.04.2017



doc. Ing. Jiří Koziorek, Ph.D.
vedoucí katedry



prof. RNDr. Václav Snašel, CSc.
děkan fakulty

Prohlášení studenta

„Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.“

V Ostravě dne 28. 4. 2017



Poděkování

Tímto bych rád poděkoval vedoucímu své bakalářské práce, doc. Ing. Radku Martinkovi, Ph.D., za odbornou pomoc, podnětné rady a čas při vedení bakalářské práce.

Abstrakt

Bakalářská práce pojednává o problematice zpracování řečových signálů a adaptivní filtrace. V této práci je zpracována literární řešerše problematiky řečových signálu v časové a frekvenční oblasti. V práci je realizována aplikace v online i offline režimu v programu LabVIEW pro ověření adaptivního zpracování. Program v online režimu slouží pro tvorbu databáze nahrávek řeči a různých rušení z venkovním prostředí, tak i z domácnosti. Aplikace v offline režimu pak slouží pro testování adaptivních algoritmů. Kritériem pro porovnání účinnosti jednotlivých algoritmů je především zvýšení poměru signálu k šumu. Pro testování míry filtrace byla zvolena metoda globálního SNR.

Abstract

The bachelor thesis deals with problems of speech processing and adaptive filtration. In this work the literary research of the speech signal in the time and frequency domain is processed. The work is implemented in both online and offline mode in LabVIEW to verify adaptive processing. The online mode program is used to create a database of speech recordings and various interferences from the outdoor environment as well as from the home. The offline application then serves to test adaptive algorithms. The criterion for comparing the efficiency of individual algorithms is primarily to increase the signal to noise ratio. To test the filtration rate, a global SNR method was chosen.

Klíčová slova

Adaptivní filtry, adaptivní algoritmy, LabVIEW, zpracování řeči, potlačení šumu.

Key words

Adaptive Filters, Adaptive Algorithms, LabVIEW, Speech Processing, Noise Reduction.

Obsah

1	Analýza řečového signálu	15
1.1	Kódování tvaru vlny.....	15
1.1.1	Pulsní kódová modulace.....	15
2	Zpracování řečových signálů v časové a frekvenční oblasti	18
2.1	Zpracování v časové oblasti	18
2.1.1	Váhové okénko.....	18
2.1.2	Krátkodobá energie	19
2.1.3	Krátkodobá funkce středního počtu průchodu signálu nulou.....	20
2.1.4	Krátkodobá autokorelační funkce	21
2.2	Zpracování ve frekvenční oblasti	22
2.2.1	Krátkodobá Fourierova transformace.....	22
2.3	Lineární prediktivní analýza.....	23
2.4	Kepstrální analýza mluvené řeči	24
2.4.1	Kepstrum	24
3	Popis měřicích mikrofónů G.R.A.S. 40PP a karty NI 9234	26
3.1	Měřicí mikrofony G.R.A.S. 40PP	26
3.2	Měřicí karta NI 9234	27
4	Adaptivní filtrace.....	28
4.1	Lineární transversální FIR filtr.....	28
4.2	Wienerův filtr	29
4.3	Charakteristika adaptivních algoritmů	30
4.3.1	Algoritmus LMS	30
4.3.2	Algoritmus NLMS.....	31
4.3.3	Algoritmus RLS	31
4.3.4	Algoritmus QR-RLS	32
5	Praktická realizace vlastního systému pro adaptivní filtraci	33
5.1	Popis realizovaného programu v offline režimu	33
5.1.1	Architektura producent-konzument.....	33
5.1.2	Algoritmus offline aplikace.....	34
5.2	Změna vlastností filtru vlivem změny parametrů.....	39
5.2.1	Vliv délky filtru.....	39

5.2.2	Vliv konvergenční konstanty.....	42
5.3	Efektivita filtrace adaptivních algoritmů.....	45
5.3.1	Objektivní kritérium posouzení kvality filtračního procesu.....	47

Seznam použitých zkratk

PCM – pulsně šířková modulace

LPC – lineární prediktivní kódování

A/D – převod analogového (spojitého) signálu na digitální (diskrétní)

AC/DC – střídavý/stejnoseměrný proud

FIR – konečná impulsní odpověď (Finite Impulse Response)

LMS – střední kvadratická odchylka (Last Mean Square)

NLMS – normalizovaný LMS algoritmus

RLS – rekurzivní adaptivní filtr

QR-RLS – rekurzivní adaptivní filtr s QR rozkladem

VAD – detektor řeči

GSNR – globální odstup signálu od šumu (Global Signal to Noise Ratio)

dB – decibel

Seznam použitých symbolů

$s(t)$ – signál spojitý v čase

$s(k)$ – značí vzorek akustického signálu

T – perioda

F_v – vzorkovací frekvence

Q_n – krátkodobá charakteristika

$w(n)$ – značí pravoúhlé nebo Hammingovo okénko

E_n – krátkodobá energie

Z_n – krátkodobá funkce středního počtu průchodu nulou

$R_n(m)$ – krátkodobá autokorelační funkce

$S(\omega, n)$ – krátkodobá Fourierova transformace

$H(z)$ – přenosová funkce modelu

$\mathbf{w}(n)$ – vektor koeficientů adaptivního filtru

$\mathbf{x}(n)$ vektor vstupního signálu

$\mathbf{K}(n)$ – ziskový vektor

$\mathbf{P}(n)$ – inverzní autokorelační matice

$\xi(n)$ – kritérium chyby

N – délka segmentu

σ_s^2 – výkon řečového signálu

σ_n^2 – výkon aditivního šumu

Seznam obrázků

Obrázek 1: Ilustrace procesu vzorkování.	16
Obrázek 2: Ilustrace procesu kvantování.	17
Obrázek 3: Pravoúhlé okénko v časové (vlevo) a frekvenční oblasti (vpravo).	19
Obrázek 4: Hammingovo okénko v časové (vlevo) a frekvenční oblasti (vpravo).	19
Obrázek 5: Zobrazení krátkodobé energie řečového signálu.	20
Obrázek 6: Zobrazení krátkodobé funkce středního počtu průchodu nulou řečového signálu.	21
Obrázek 7: Zobrazení krátkodobé Autokorelační funkce znělé hlásky "á".	22
Obrázek 8: Zobrazení spektrogramu pomocí krátkodobé Fourierovy transformace.	23
Obrázek 9: Model vytváření řeči s lineárním číslicovým filtrem.	23
Obrázek 10: Model výpočtu kepstra.	24
Obrázek 11: Zobrazení průběhu znělé hlásky "á".	25
Obrázek 12: Zobrazení kepstra znělé hlásky "á".	25
Obrázek 13: Měřicí mikrofon G.R.A.S. 40PP [12].	26
Obrázek 14: Měřicí karta NI 9234 [13].	27
Obrázek 15: Obecné schéma adaptivního lineárního filtru.	28
Obrázek 16: Přímá (transverzální) struktura FIR filtru.	29
Obrázek 17: Blokové schéma Wienerova filtru.	29
Obrázek 18: Struktura producent-konzument.	33
Obrázek 19: Algoritmus offline aplikace.	34
Obrázek 20: Zpracování zprávy pro načtení nahrávek.	35
Obrázek 21: Blokový diagram vyhodnocení pro stejnou délku signálů.	35
Obrázek 22: Zpracování zprávy pro stejnou délku signálů.	36
Obrázek 23: Ukázka filtrace algoritmu RLS.	36
Obrázek 24: Váhování a segmentace spektrogramu.	37
Obrázek 25: Zisk výkonového spektra.	37
Obrázek 26: Ukázka vykreslení spektrogramu v offline aplikaci.	38
Obrázek 27: Čelní panel offline aplikace.	38
Obrázek 28: Průběh řečové nahrávky, šumu a šum přičtený k nahrávce.	39
Obrázek 29: Průběhy délky filtru a chyby filtru pro algoritmus LMS.	40
Obrázek 30: Průběhy délky filtru a chyby filtru pro algoritmus RLS.	41
Obrázek 31: Průběhy konvergenční konstanty pro algoritmus LMS.	43
Obrázek 32: Vliv délky filtru a konvergenční konstanty na filtraci.	44
Obrázek 33: Blokové schéma měření rušení.	45
Obrázek 34: Průběhy rušení–primární mikrofon zavřená okna (vlevo), otevřená okna (vpravo).	46
Obrázek 35: Nahrávka řeči.	47
Obrázek 36: Výskyt šumu při přenosu řeči.	47
Obrázek 37: Graf úspěšnosti filtrace, elektromobil.	48
Obrázek 38: Graf úspěšnosti filtrace, Škoda Felicie.	49
Obrázek 39: Decimace signálu.	50
Obrázek 40: Vliv převzorkování na filtraci pro algoritmus LMS, elektromobil.	51
Obrázek 41: Výsledky filtrace pro algoritmus LMS, Škoda Felicie.	52
Obrázek 42: Výsledky filtrace pro algoritmus NLMS, elektromobil.	52

Obrázek 43: Výsledky filtrace pro algoritmus NLMS, Škoda Felicie.....	53
Obrázek 44: Výsledky filtrace pro algoritmus RLS, elektromobil.....	54
Obrázek 45: Výsledky filtrace pro algoritmus RLS, Škoda Felicie.	54

Seznam tabulek

Tabulka 1: Detailní specifikace mikrofonu G.R.A.S. 40PP.	26
Tabulka 2: Technické parametry měřicí karty NI 9234.	27
Tabulka 3: Tabulka doby výpočtu pro algoritmus LMS.	41
Tabulka 4: Délka výpočtu pro algoritmus RLS.....	42
Tabulka 5: Maximální chyba a dobá ustálení pro různou délku filtru.....	44
Tabulka 6: Maximální chyba a doba ustálení pro různou délku konvergenční konstanty.	44
Tabulka 7: Výsledky pro jednotlivé filtrace, elektromobil.....	48
Tabulka 8: Výsledky pro jednotlivé filtrace, Škoda Felicie.	49
Tabulka 9: Decimální faktor přepočítaný na frekvenci.	49
Tabulka 10: Vliv převzorkování na hodnoty SNR pro vstupní signál.	50
Tabulka 11: Výsledky filtrace pro algoritmus LMS, elektromobil.	51
Tabulka 12: Výsledky filtrace pro algoritmus LMS, Škoda Felicie.....	51
Tabulka 13: Výsledky filtrace pro algoritmus NLMS, elektromobil.	52
Tabulka 14: Výsledky filtrace pro algoritmus NLMS, Škoda Felicie.	53
Tabulka 15: Výsledky filtrace pro algoritmus RLS, elektromobil.	53
Tabulka 16: Výsledky filtrace pro algoritmus RLS, Škoda Felicie.....	54
Tabulka 17: Obsah příloženého DVD.....	58

Úvod

Filtrace patří mezi základní metody číslicového zpracování signálů. Toto předzpracování výstupního signálu je důležité při odstraňování nežádoucích vnitřních či vnějších vlivů působících na snímáný signál.

Důležitým prostředkem při zpracování dat jsou také adaptivní systémy. Již z názvu vyplývá, že se jedná o systémy, které se přizpůsobují svými parametry s časem. Využívají se především při filtraci signálů, detekci užitečného signálu v šumu nebo identifikaci neznámého systému.

Obsahem této práce je jeden z mnoha typů filtrace signálu spojený se základním využitím adaptivních systémů – adaptivní filtrace. Tento druh filtrace představuje velice efektivní nástroj při zpracování zarušeného signálu, jehož výhodou je široká využitelnost při odstraňování rušení, jehož vlastnosti se s časem mění nebo jehož vlastnosti nejsou předem známy.

Bakalářská práce se skládá ze dvou hlavních částí – teoretické a praktické. V první z nich je představen teoretický úvod představované problematiky. První a druhá kapitola se zabývá řešerší problematiky zpracování řečových signálů v časové a frekvenční oblasti. Jedná se především o zpracování signálu ze spojitě do diskrétní podoby, dále popis řešení v časové oblasti pomocí krátkodobé analýzy a ve frekvenční oblasti pomocí krátkodobé Fourierovy transformace. V další kapitole je obsažen technický popis mikrofónů a měřicí karty, pomocí nichž byly pořízeny nahrávky v programu LabVIEW. Čtvrtá kapitola popisuje základní vlastnosti adaptivních algoritmů, které jsou obsaženy v knihovně NI LabVIEW Adaptive Filter Toolkit. Poslední kapitola popisuje ověření funkčnosti jednotlivých adaptivních algoritmů, chování parametrů a jejich optimálního nastavení.

Cílem bakalářské práce je zpracování řečových signálů a adaptivní filtrace v online režimu, tedy zpracování v reálném čase v časové i frekvenční oblasti, filtrace a ukládání naměřených dat. Následně je možné naměřená data analyzovat v offline aplikaci pro podrobnější analýzu a pochopení problematiky.

1 Analýza řečového signálu

Zpracování řečového signálu je jednou z nejdůležitějších součástí všech systémů v oblasti rozpoznávání mluvené řeči, počítačové syntézy, identifikaci a verifikaci člověka na základě jeho hlasu či pro efektivní kódování nebo přenos řečového signálu. Požadavky kladené na takové systémy se značně liší. Například přenosová rychlost bývá rozhodujícím parametrem pro účely efektivního kódování či přenosu řečového signálu. Jiná hlediska je třeba sledovat v úlohách rozpoznávání mluvené řeči, kde rozhodujícím faktorem je kvalita informačního obsahu řečové reprezentace. Naopak u počítačové syntézy je hlavní dokonalá srozumitelnost a vysoká přirozenost syntetizované řeči. U identifikace a verifikace řečníka rozhoduje parametrizace, která posílí a zvýrazní odlišnosti hlasů.

Základním předpokladem většiny metod analýzy akustického signálu řeči je ten, že se jeho vlastnosti v průběhu času mění pomalu. Tento předpoklad vede na metody krátkodobé analýzy, při kterých se signál rozdělí na krátké časové úseky (mikrosegmenty) o typické délce 10 ms. Takto vzniklé oddělené krátké zvuky jsou popsány číslem nebo souborem čísel. Protože mikrosegmenty na sebe navazují, dostaneme výsledné časové číselné posloupnosti, které jsou výsledkem analýzy a popisují promluvený celek. Metody krátkodobé analýzy používají většinou jako vstup data získaná digitalizací signálu, tj. kódováním tvaru vlny [1].

1.1 Kódování tvaru vlny

Pro zdrojové kódování řečových signálů se používají kodéry tvarového průběhu realizující tzv. zdrojové kódování tvaru vlny. Jejich principem je, aby se časový průběh analogového signálu na výstupu dekodéru co nejvíce shodoval s časovým průběhem analogového signálu na vstupu kodéru. K nejjednodušším způsobům tohoto kódování patří pulsní kódová modulace PCM.

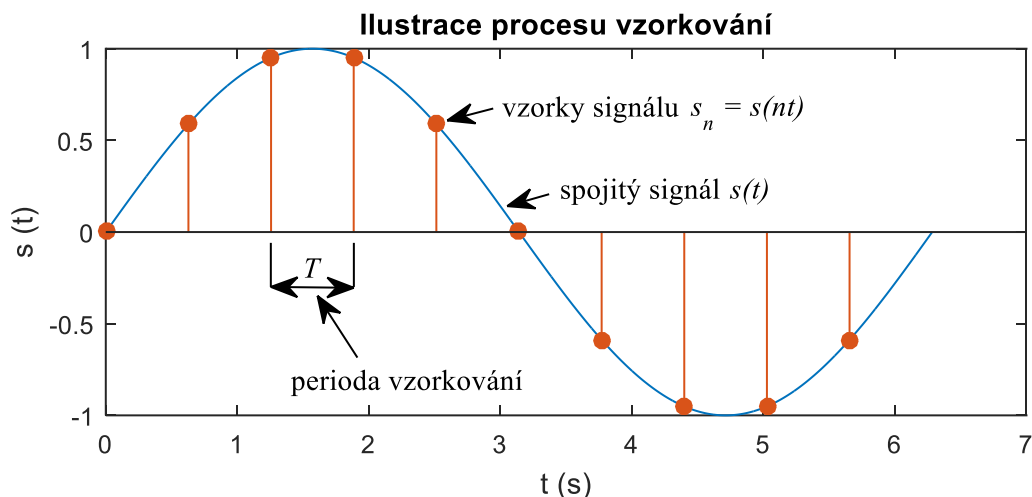
1.1.1 Pulsní kódová modulace

Mluvená řeč obsahuje informace, které jsou obvykle získávány zpracováním zvukových kmitů řečového signálu snímaného mikrofonom [1]. Princip PCM nebo též digitalizace spočívá v pravidelném odečítání hodnoty signálu pomocí analogově – digitálního A/D převodníku a jeho signálu v binární podobě. Tento proces zahrnuje provedení dvou kroků, a to vzorkování a kvantování s kódováním.

1.1.1.1 Vzorkování

Vzorkování je transformace signálu $s(t)$ spojitého v čase na posloupnost vzorků $s_n = s(nT)$, kde T je perioda vzorkování a $n \in \langle 0, \infty \rangle$, viz Obrázek 1. Frekvence vzorkování $F_v = 1/T$ je v souladu s Nyquistovým vzorkovacím teorémem omezena, a musí splňovat $F_v \geq 2F_m$, kde F_m je horní hranice frekvenčního pásma analogového signálu $s(t)$ a dolní hranice frekvenčního pásma je 0 Hz [1]. Analogový signál $s(t)$ je definován rovnicí (1.1).

$$s(t) = \sum_{n=-\infty}^{\infty} s(nT) \left[\frac{\sin \pi \left(\frac{t}{T} - n \right)}{\pi \left(\frac{t}{T} - n \right)} \right]. \quad (1.1)$$



Obrázek 1: Ilustrace procesu vzorkování.

1.1.1.2 Kvantování a kódování

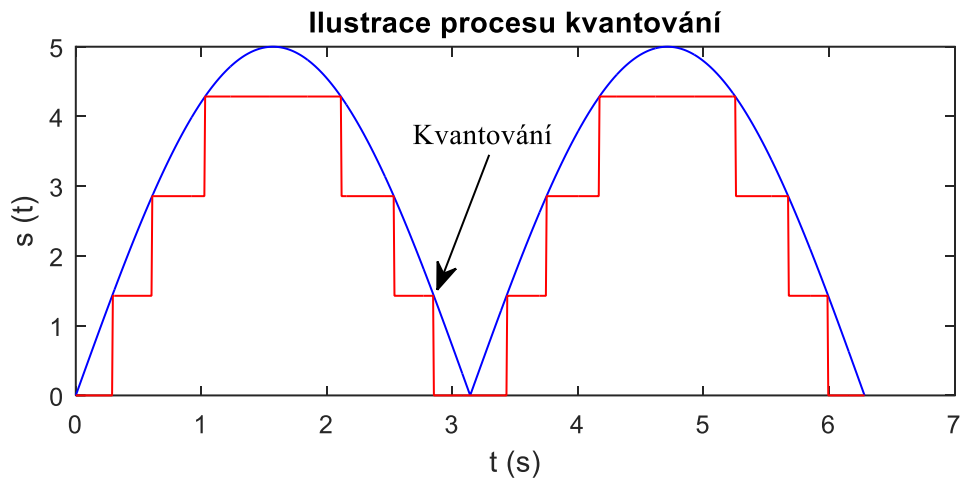
Kvantování (viz Obrázek 2) s následujícím kódováním je aproximací analogové hodnoty vzorku signálu jednou z konečného počtu číselných hodnot. Děje se tak pomocí A/D převodníku, který zajišťuje převod analogového signálu na disktrétní. Tento převodník má několik důležitých vlastností:

- Rozlišovací schopnost – je dána počtem rozlišitelných úrovní kvantovaného analogového signálu. U profesionálních zvukových karet se lze setkat s rozlišovací schopností 24bitů, u běžných se používá rozlišení 16bitů, což odpovídá CD kvalitě zvuku.
- Kvantizační krok – neboli citlivost. Jedná se o rozdíl napětí dvou vstupních analogových hodnot, kdy dojde k přechodu z jednoho digitálního výstupu na druhý.
- Chyba kvantování – jedná se o maximální hodnotu rozdílu mezi analogovým vstupním napětím a výstupní digitální hodnotou napětí. Většinou odpovídá polovině kvantizačního kroku.

Pro návrh kvantizéru, který má rovnoměrně rozložené kvantizační úrovně, je dostačující udat počet úrovní kvantování a kvantizační krok Δ . Počet úrovní se obvykle volí ve tvaru 2^B (B je počet bitů v binárním kódu) a parametry Δ a B se vybírají tak, aby byl pokryt celý rozsah signálu [1]. Budeme-li předpokládat, že $|s(nT)| \leq S_{max}$, kde S_{max} je maximální úroveň vzorkovaného signálu, dostaneme rovnici (1.2):

$$2S_{max} = \Delta 2^B. \quad (1.2)$$

Při procesu kvantování dochází k „zaokrouhlování“ okamžitých velikostí signálu. Toto zaokrouhlení má ztrátový účinek, kdy je hodnota zaokrouhlena na nejbližší nižší celočíselnou hodnotu. Tato ztráta se nazývá kvantizační zkreslení nebo kvantizační šum.



Obrázek 2: Ilustrace procesu kvantování.

Kvantizační proces je nevratný. To znamená, že informace o původním tvaru signálu jsou ztraceny, když je daný signál nakvantován na určité hladiny. Velikost kvantizačního kroku má vliv na tvar výsledného signálu. Pokud bude příliš velký dojde ke zkreslení informace obsažené v signálu.

2 Zpracování řečových signálu v časové a frekvenční oblasti

V této kapitole je popsána problematika zpracování řečových signálu v časové a frekvenční oblasti. Jedná se o popis řešení v časové oblasti pomocí krátkodobé analýzy a ve frekvenční oblasti pomocí krátkodobé Fourierovy transformace. Dále je popsána Lineární prediktivní analýza a kepsrální analýza mluvené řeči.

2.1 Zpracování v časové oblasti

Většinu metod krátkodobé analýzy v časové oblasti lze vyjádřit rovnicí (2.1):

$$Q_n = \sum_{k=-\infty}^{\infty} \tau(s(k))w(n-k), \quad (2.1)$$

kde Q_n je krátkodobá charakteristika, $s(k)$ značí vzorek akustického signálu získaný pulsně šířkovou modulací v čase k , $\tau(\cdot)$ vyjadřuje příslušnou transformační funkci a $w(n)$ je váhová posloupnost neboli tzv. okénko, kterým se vybírají, respektive váží vzorky $s(k)$ [1].

2.1.1 Váhové okénko

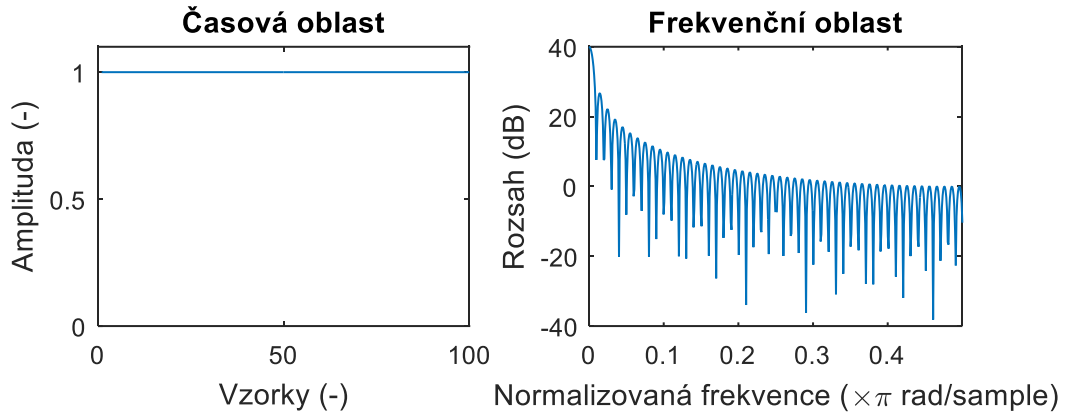
Po rozhodnutí pro určitou velikost mikrosegmentu je implicitně předpokládáno, že zvukový signál je v okolí okénka periodický s periodou okénka. Chyba, která vzniká nesouladem s tímto předpokladem, může být do jisté míry kompenzována použitím tzv. okénka. Okénko je posloupnost vah pro prvky mikrosegmentu. Nejběžnější používané typy okének jsou pravoúhlé a Hammingovo.

2.1.1.1 Pravoúhlé okénko

U pravoúhlého okénka (viz Obrázek 3) je aplikována stejná váha na všechny okénkem vybrané vzorky [1]. Okénko je definováno vztahem (2.2):

$$w(n) = \begin{cases} 1 & \text{pro } 0 \leq n \leq L-1 \\ 0 & \text{pro ostatní } n, \end{cases} \quad (2.2)$$

kde L je počet vzorků vybraných okénkem.

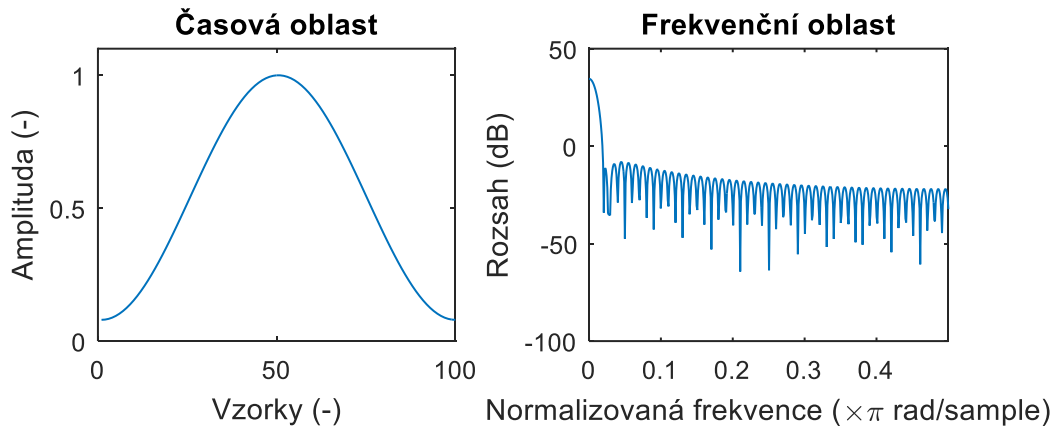


Obrázek 3: Pravoúhlé okénko v časové (vlevo) a frekvenční oblasti (vpravo).

2.1.1.2 Hammingovo okénko

V častých případech, kdy je při zpracování potřeba potlačit vzorky na krajích okénka je vhodné použít Hammingovo okénko (viz Obrázek 4), které je definováno vztahem (2.3):

$$w(n) = \begin{cases} 0,54 - 0,46 \cos\left(\frac{2\pi n}{L-1}\right) & \text{pro } 0 \leq n \leq L-1 \\ 0 & \text{pro ostatní } n. \end{cases} \quad (2.3)$$



Obrázek 4: Hammingovo okénko v časové (vlevo) a frekvenční oblasti (vpravo).

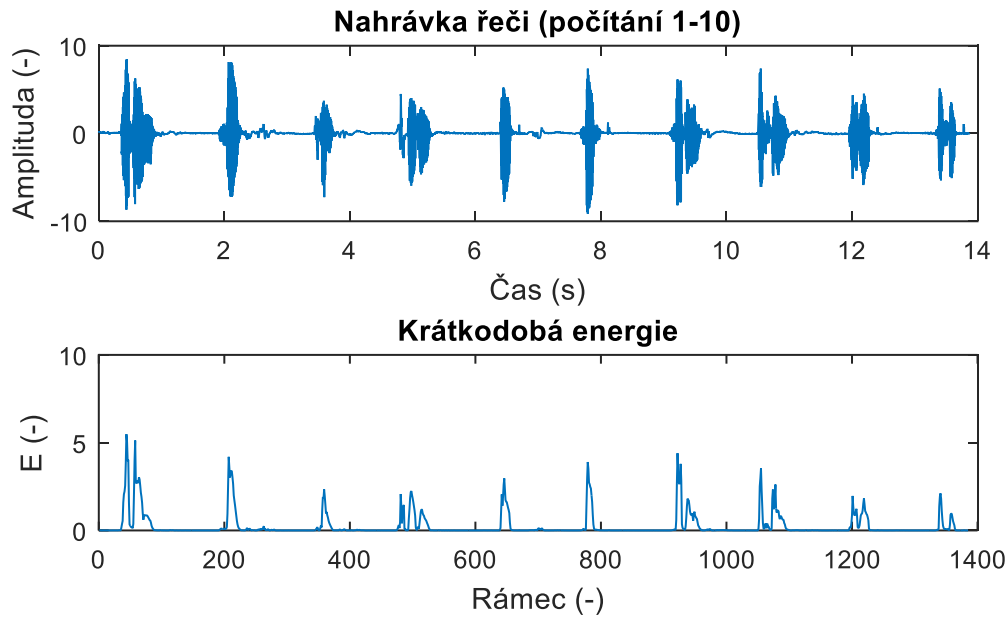
2.1.2 Krátkodobá energie

Funkci krátkodobé energie signálu lze definovat vztahem (2.4):

$$E_n = \sum_{k=-\infty}^{\infty} [s(k)w(n-k)]^2, \quad (2.4)$$

kde $s(k)$ je vzorek signálu v čase k a $w(n)$ reprezentuje příslušný typ okénka. Při měření krátkodobé energie je vhodné volit délku mikrosegmentu shodnou s délkou okénka, a to o velikosti 10-20 ms při frekvenci vzorkování odpovídající kvalitě zpracovávaného signálu. Výsledkem krátkodobé energie jsou

pro každý mikrosegment hodnoty udávající průměrnou energii v daném mikrosegmentu, viz Obrázek 5 [1].



Obrázek 5: Zobrazení krátkodobé energie řečového signálu.

2.1.3 Krátkodobá funkce středního počtu průchodu signálu nulou

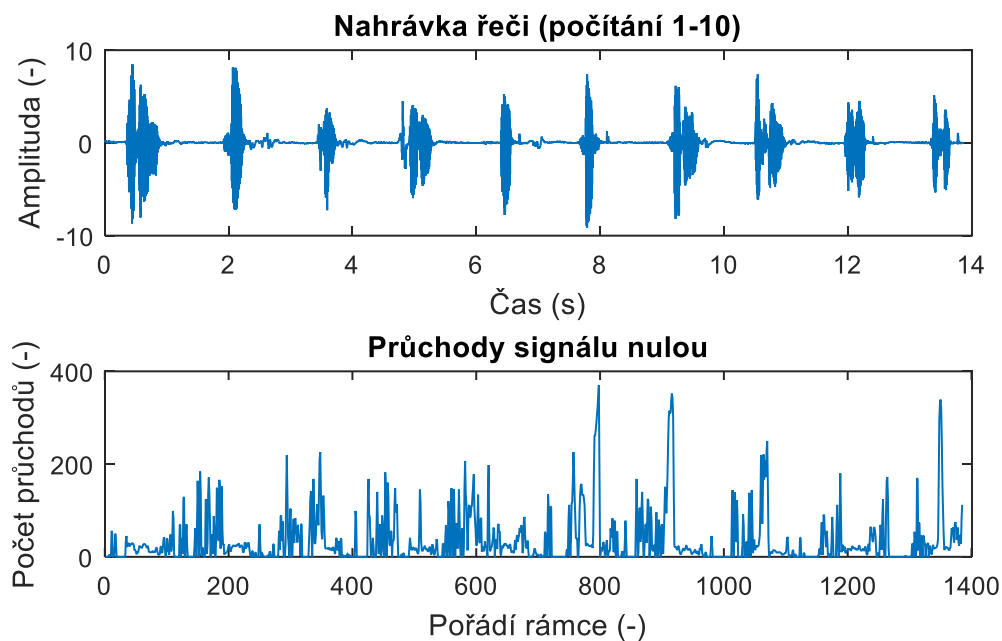
Na rozdíl od charakteristiky krátkodobé energie, zaměřené na sledování amplitudy signálu, poskytuje krátkodobá funkce středního počtu průchodu signálu nulou (viz Obrázek 6) informaci o frekvenčních vlastnostech signálu. Krátkodobou funkci středního počtu průchodů nulou lze definovat rovnicí (2.5):

$$Z_n = \sum_{k=-\infty}^{\infty} |\text{sgn}[s(k)] - \text{sgn}[s(k-1)]| w(n-k), \quad (2.5)$$

kde

$$\text{sgn}[s(k)] = \begin{cases} 1 & \text{pro } s(k) \geq 0 \\ -1 & \text{pro } s(k) < 0, \end{cases} \quad (2.6)$$

a $w(n)$ je pravoúhlé okénko. Metody krátkodobé funkce středního počtu průchodu nulou jsou vhodné v procesech, kde se určují začátky a konce promluv pro určení základního hlasivkového tónu nebo zjištění přibližné hodnoty frekvence nejsilnějšího formantu. Nevýhodou těchto metod je nízká přenosová rychlost zpracování (800 bit/s), a také ani zde nelze z hodnot charakteristiky rekonstruovat původní signál [1].



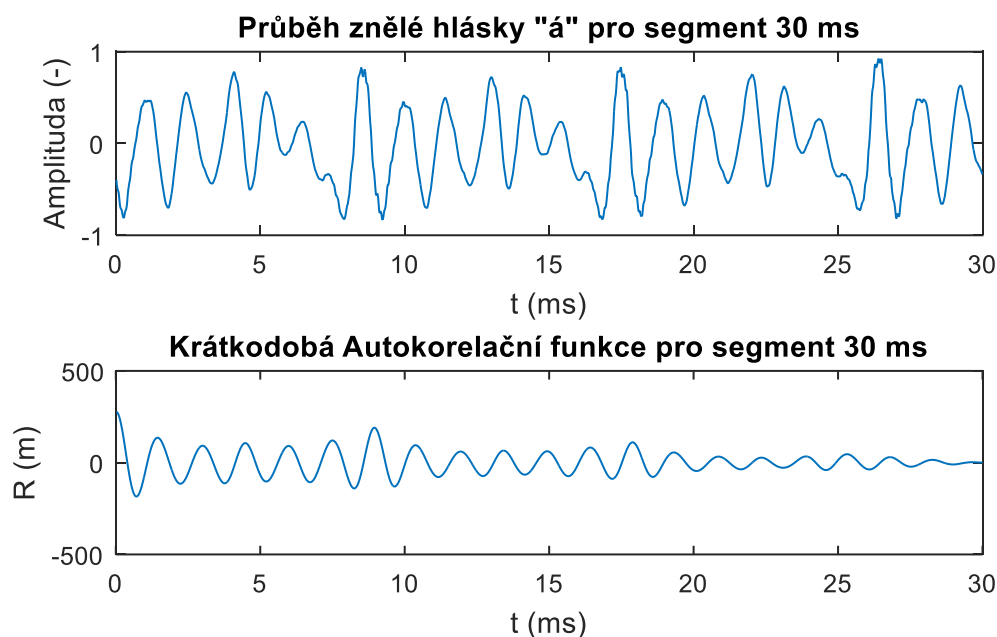
Obrázek 6: Zobrazení krátkodobé funkce středního počtu průchodu nulou řečového signálu.

2.1.4 Krátkodobá autokorelační funkce

Krátkodobá autokorelační funkce je definována vztahem (2.7):

$$R_n(m) = \sum_{k=-\infty}^{\infty} s(k)w(n-k)s(k+m)w(n-k-m), \quad (2.7)$$

kde $w(n)$ je opět pravoúhlé nebo Hammingovo okénko. Autokorelační funkce se využívá zejména při indikaci periodičnosti signálu. Jestliže je totiž zpracováváný signál periodický s periodou T , pak autokorelační funkce nabývá maximální hodnoty právě pro $m = 0, T, 2T, \dots$. Z uvedeného důvodu je charakteristika velmi vhodná například pro určování periody základního hlasivkového tónu, viz Obrázek 7 [1].



Obrázek 7: Zobrazení krátkodobé Autokorelační funkce znělé hlásky "á".

2.2 Zpracování ve frekvenční oblasti

Mluvená řeč může být ve frekvenční oblasti reprezentována jako kompozice spektrální obálky charakterizující vlastnosti hlasového ústrojí a jemné struktury charakterizující buzení. Jak se mění buzení a tvar hlasového ústrojí, vytvářejí se rozdílné zvuky a mění se i spektrum signálu, viz Obrázek 8. Nejčastěji používané postupy jsou založeny na aplikaci krátkodobé Fourierovy transformace [1].

2.2.1 Krátkodobá Fourierova transformace

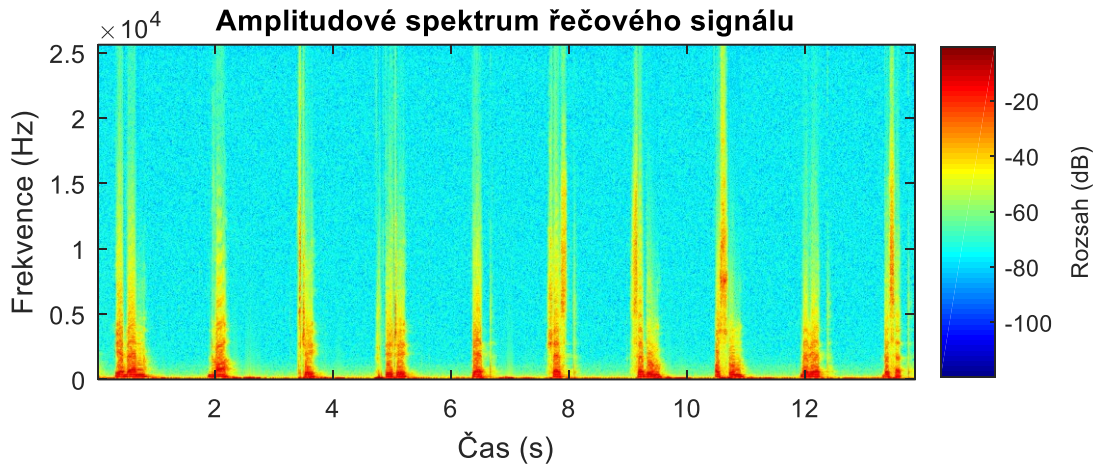
Předpokládejme, že metodou PCM byly získány vzorky $s(k)$ řečového signálu. Krátkodobá Fourierova transformace $S(\omega, n)$ je pak definována vztahem (2.8):

$$S(\omega, n) = \sum_{k=-\infty}^{\infty} s(k)h(n-k)e^{-j\omega k}, \quad (2.8)$$

kde $h(n)$ je zatím blíže nespecifikovaná funkce okénka, která vybírá pro zpracování určený úsek signálu. Je zřejmé, že takto vyjádřený Fourierův obraz je funkcí spojitě proměnné frekvence ω , tak i diskrétní proměnné času n a odpovídá v podstatě konvoluci okénka $h(n)$ a $s(n)$ modulovaného $e^{-j\omega n}$ [1]. Po odvození dostaneme rovnici (2.9):

$$S(\omega, n) = [s(n)e^{-j\omega n}] * h(n), \quad (2.9)$$

kde operace $*$ označuje konvoluci.



Obrázek 8: Zobrazení spektrogramu pomocí krátkodobé Fourierovy transformace.

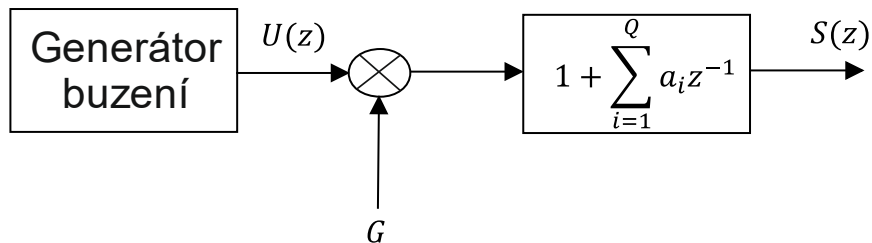
2.3 Lineární prediktivní analýza

Lineární prediktivní kódování (LPC) je jednou z neefektivnějších metod analýzy akustického signálu. Tato metoda je opět založena na krátkodobém odhadu, kdy se snaží přímo z řečového signálu získat parametry modelu vytváření řeči. Model vytváření řeči je zobrazen na (Obrázek 9). Její vysoká efektivnost spočívá v zabezpečení odhadu parametrů při relativně přijatelné výpočetní zátěži. Princip LPC je založen na předpokladu, že k -tý vzorek signálu $s(k)$ lze popsat lineární kombinací Q předchozích vzorků a buzení $u(k)$ [1]. Z tohoto předpokladu dostaneme rovnici (2.10):

$$s(k) = \sum_{i=1}^Q a_i s(k-i) + Gu(k), \quad (2.10)$$

kde G je koeficient zesílení a Q je řád modelu. Přenosovou funkci modelu $H(z)$ lze pak zapsat ve tvaru (2.11):

$$H(z) = \frac{S(z)}{U(z)} = \frac{G}{A(z)} = \frac{G}{1 + \sum_{i=1}^Q a_i z^{-1}}. \quad (2.11)$$



Obrázek 9: Model vytváření řeči s lineárním číslicovým filtrem.

2.4 Kepstrální analýza mluvené řeči

Kepstrální analýza byla poprvé publikována v roce 1963 a lze ji zařadit do tzv. homomorfického, obecně nelineárního zpracování signálů. V aplikaci na řečové signály vychází z časově-frekvenční analýzy [4]. Řečový signál je proměnný v čase a chceme-li zachytit tuto změnu ve spektrální oblasti, je zapotřebí analyzovat jen segment takové délky, kde lze předpokládat stacionaritu signálu.

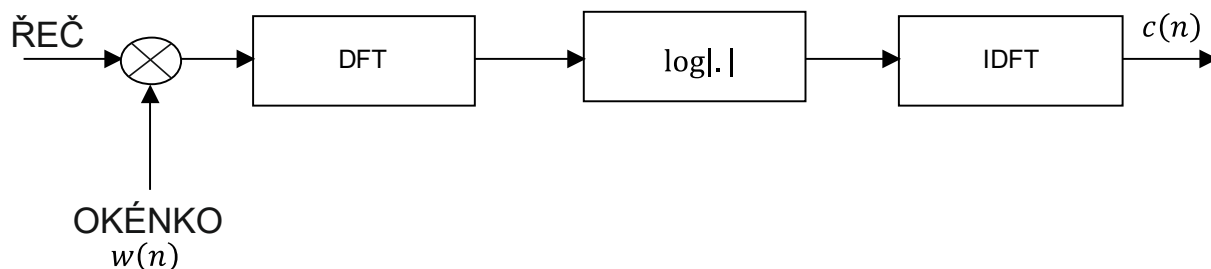
Tyto analýzy jsou vhodné v metodách pro oddělování signálů vzniklých násobením více složek nebo konvolucí. Výsledky jsou užitečné pro rozpoznávání řeči nebo rozpoznávání mluvčích a jejich stavů. Proto je použití této techniky pro analýzu a parametrizaci řečového signálu výhodné a efektivní [15].

Hlasivky vytvářejí kvazi-periodickou nebo šumovou budící funkci a hlasový trakt s impulsní odezvou tuto funkci moduluje. Na výstupu hlasového traktu vzniká řečový signál konvolucí, což v kmitočtové oblasti odpovídá násobení obou funkcí po Fourierově transformaci.

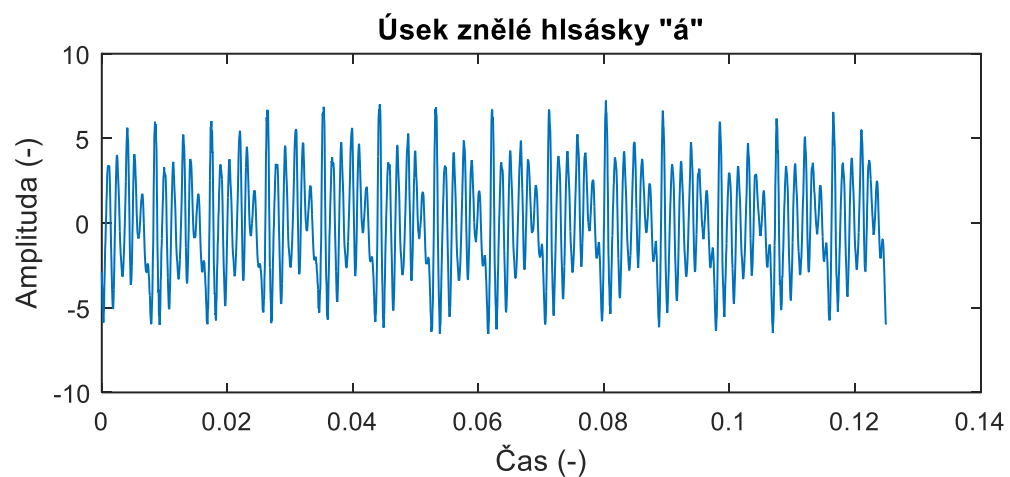
2.4.1 Kepstrum

Slovo „kepstrum“ je přesmyčkou ze slova spektrum, kdy se vymění pořadí prvních čtyř písmen. Samotná funkce je definována jako absolutní hodnota inverzní Fourierovy transformace logaritmu amplitudového spektra signálu [1].

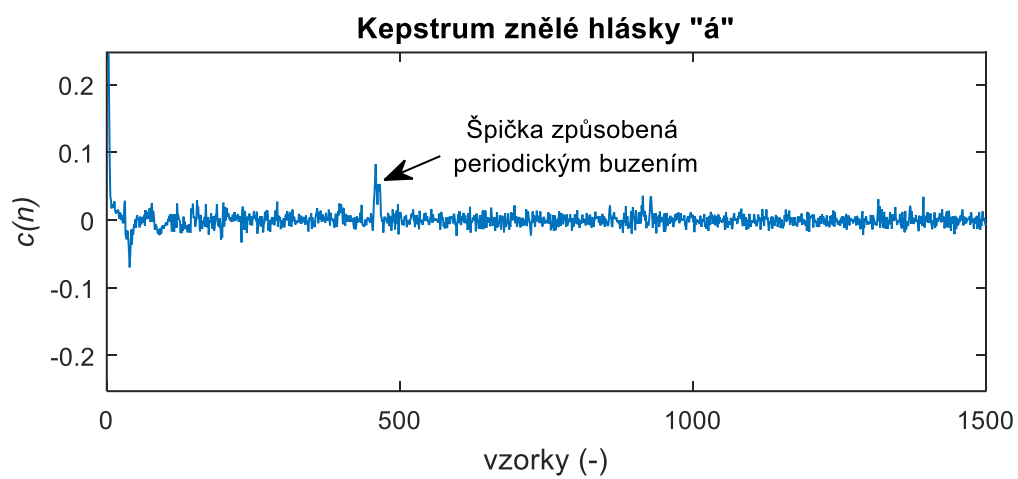
Proces kepstrální analýzy je znázorněn na (Obrázek 10). Na vstup je přiveden signál, který vznikl diskrétní konvolucí $s(n)$ a funkce okénka $w(n)$ například Hammingova. Tento signál je přiveden na vstup bloku DFT. Následně do bloku $\log|\cdot|$ přichází signál, který je roven Fourierovy transformace buzení a impulsní odezvy hlasového ústrojí [1]. Zobrazení kepstra je na (Obrázek 12).



Obrázek 10: Model výpočtu kepstra.



Obrázek 11: Zobrazení průběhu znělé hlásky "á".



Obrázek 12: Zobrazení kepra znělé hlásky "á".

3 Popis měřicích mikrofونů G.R.A.S. 40PP a karty NI 9234

V této kapitole je popsána hardwarová specifikace měřicích mikrofونu G. R. A.S. 40PP a karty NI9234. Tento hardware byl použit pro měření řečových signálů a šumu, a také pro online analýzu adaptivních filtrů.

3.1 Měřicí mikrofony G.R.A.S. 40PP

Měřicí mikrofony typu G. R. A. S 40PP (viz Obrázek 13) se hojně využívají pro vícekanálová měření (analýzy), testování přijímačů a ovladačů nebo mikroreproduktorů.

Frekvenční rozsah těchto mikrofونů je 10 Hz až 20 kHz, dynamický rozsah 30 dB až 135 dB a citlivost 50 mV/Pa. Takové parametry činí tyto mikrofony velmi citlivými na okolní hluk. Při odstupu pouhého jednoho metru od mikrofону je snímán signál podstatně horší. To je způsobeno právě velkým dynamickým rozsahem, který snímá jak promluvu řečníka, tak okolní hluk. Tyto mikrofony proto nejsou příliš vhodné pro širokopásmová měření.

Výstupem mikrofону je standardní BNC konektor pro používání kabelů RG58, které slouží k propojení s měřicí kartou NI 9234.

Tabulka 1: Detailní specifikace mikrofону G.R.A.S. 40PP.

Technické parametry	Rozsahy
Frekvenční rozsah (± 1 dB)	20 Hz až 20 kHz
Frekvenční rozsah (± 2 dB)	10 Hz až 20 kHz
Spodní limit dynam. rozsahu	< 30 dB
Horní limit – dynam. rozsahu	135 dB
Nastavená citlivost	50 mV/Pa
Polarizační napětí	0 V
Napájení	2-20 mA
Odvzdušnění mikrofону	Přední
Výstupní impedance	<50 Ω
Provozní teplotní rozsah	-10 až 50 / 14 až 122 $^{\circ}\text{C}$ / $^{\circ}\text{F}$
Teplotní rozsah-skladování	-40 až 85 / -40 až 185 $^{\circ}\text{C}$ / $^{\circ}\text{F}$



Obrázek 13: Měřicí mikrofون G.R.A.S. 40PP [12].

3.2 Měřicí karta NI 9234

Jak už bylo zmíněno v kapitole 3.1, karta NI 9234 (viz Obrázek 14) obsahuje 4 BNC konektory pro připojení z měřicími mikrofony. Uvnitř karty se nachází delta-sigma A/D převodník s rozlišením 24 bitů. Tyto převodníky se vyznačují jednoduchým hardwarem a složitým softwarem pracující jako signálový procesor. Obsahují integrátor, komparátor a číslicový filtr. Výhodou těchto převodníků je velká linearita a podstatně nižší nároky na vstupní (omezovací) a výstupní (rekonstrukční) analogový filtr. Základní nevýhodou sigma-delta převodníků je relativně nízká vstupní frekvence vzorkování – řádově 100 kHz.

Karta obsahuje čtyři vstupní kanály, které mohou současně digitalizovat signály při rychlostech až 51,2 kHz na kanál s vestavěnými anti-aliasingovými filtry, které se automaticky přizpůsobí vzorkovací frekvenci.

Tabulka 2: Technické parametry měřicí karty NI 9234.

Technické parametry	Rozsahy
Počet kanálů	4 analogové vstupní kanály
ADC rozlišení	24 bitů
Typ ADC	Delta-sigma (s analogový před filtrováním)
Režim vzorkování	Simultánní
Vnitřní časová základna (f_M)	
Frekvence	13,1 MHz
Přesnost	± 50 ppm max.
Rozsah rychlosti přenosu dat (f_s)	
Minimum	1652 Hz
Maximum	51200 Hz



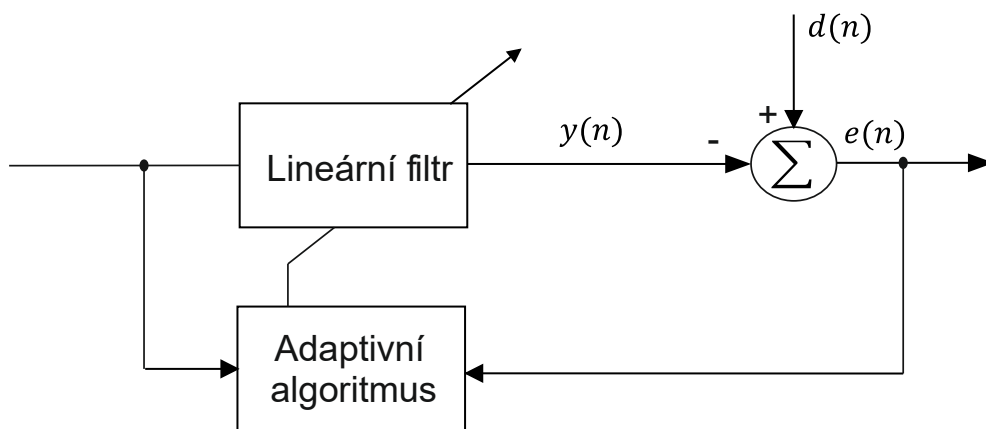
Obrázek 14: Měřicí karta NI 9234 [13].

4 Adaptivní filtrace

Hlavní myšlenka všech adaptivních filtrů spočívá v kontinuální změně parametrů podle předem stanoveného algoritmu s využitím jak informace ze vstupu, tak i pomocné informace přicházející většinou z výstupu celého systému. Adaptivní filtry využívají tzv. adaptivní algoritmy pro iterativní změnu hodnot vektoru impulzní odezvy tak, aby minimalizovaly hodnoty tzv. chybového signálu $e(n)$, který je dán rovnicí (4.1):

$$e(n) = d(n) - y(n), \quad (4.1)$$

kde $d(n)$ je požadovaný signál a $y(n)$ označuje aktuální výstup adaptivního filtru. Obecné schéma adaptivního lineárního filtru je zobrazeno na (Obrázek 15).

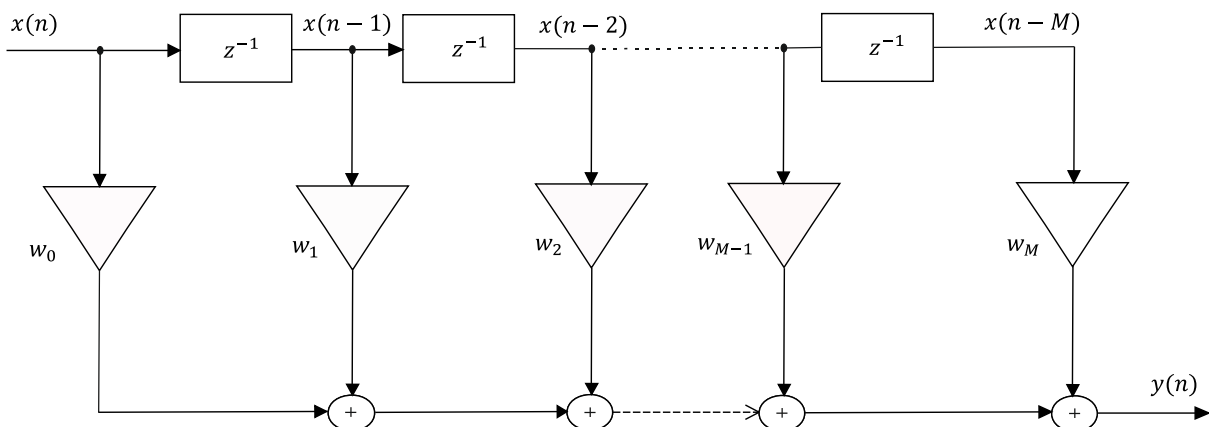


Obrázek 15: Obecné schéma adaptivního lineárního filtru.

Ve většině případů se používají lineární FIR filtry s transversální strukturou, a to z důvodů jednoduchosti implementace a nenáročnosti matematického aparátu [2].

4.1 Lineární transversální FIR filtr

Filtry s konečnou impulsovou odezvou (angl. Finite impuls response – FIR) tvoří nedílnou součást číslicového zpracování signálu. Základní, tzv. přímá struktura tohoto filtru je uvedena na (Obrázek 16). Tento typ struktury je používán ve všech testovaných adaptivních filtrech.



Obrázek 16: Přímá (transverzální) struktura FIR filtru.

Soustava je popsána rovnicí (4.2):

$$y(n) = w_0x(n) + w_1x(n-1) + w_2x(n-2) + \dots + w_Mx(n-M), \quad (4.2)$$

kde $x(n)$ je vstupní signál, $w_0, w_1 \dots w_M$ jsou váhy filtru (koeficienty) a M je délka filtru. V odborné literatuře bývá také použit pojem řád filtru N , který je definován jako $N = M - 1$.

Pro optimalizaci koeficientů adaptivního filtru můžeme stanovit vektor vah $\mathbf{w}(n)$ a vektor vstupního signálu $\mathbf{x}(n)$, které jsou zapsány následovně:

$$\mathbf{w}(n) = [w_0(n)w_1(n)w_2(n) \dots w_M(n)]^T, \quad (4.3)$$

$$\mathbf{x}(n) = [x(n)x(n-1)x(n-2) \dots x(n-M)]^T. \quad (4.4)$$

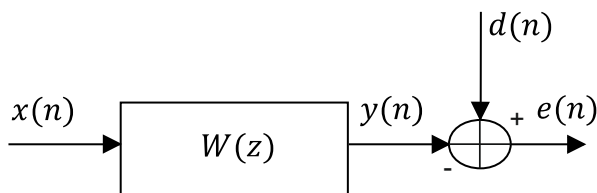
Pak výstupní signál z rovnice (4.2) lze vyjádřit jako skalární součin impulsní odezvy vektoru vah z rovnice (4.3) a vstupního vektoru z rovnice (4.4):

$$y(n) = \mathbf{w}(n)\mathbf{x}(n). \quad (4.5)$$

4.2 Wienerův filtr

Adaptivní filtry jsou do jisté míry podobné stacionárnímu Wienerovu filtru (viz Obrázek 17), avšak mohou měnit své koeficienty v čase tak, aby dokázaly efektivně potlačit nežádoucí změny na vstupním signálu [3].

Wienerův filtr klade požadavek stacionarity jak na šum, tak řečový signál. Návrh Wienerova filtru vychází ze střední kvadratické odchylky mezi originálním řečovým signálem a odhadem tohoto signálu. Odhad původního řečového signálu je výstupem Wienerova filtru [1], [4].



Obrázek 17: Blokové schéma Wienerova filtru.

Požadovaný signál je zde $d(n)$, chybový signál je $e(n)$. Vstupem filtru je signál $x(n)$ a výstupem $y(n)$. Tento výstup je generován jako lineární kombinace zpožděných vzorků výstupního signálu $y(n)$. Chyba $e(n)$ podmiňuje výběr koeficientů w_M a funkce, která tento výběr provádí by měla splňovat požadavek jednoduše vypočitatelného minima, resp. maxima, aby nedocházelo k nejednoznačnostem při konvergenci filtru. Tyto podmínky splňují právě FIR filtry s kritériem chyby ξ [3]. Kritérium chyby je definováno vztahem (4.6):

$$\xi(n) = E[e^2(n)] = E[(d(n) - y(n))^2], \quad (4.6)$$

kde $E[\]$ vyjadřuje střední očekávanou hodnotu chyby.

4.3 Charakteristika adaptivních algoritmů

Cílem adaptivních algoritmů je dosáhnout co možná nejlepšího odhadu koeficientů filtru, aby se výstupní signál co nejvíce přibližoval k signálu vstupnímu. Při procesu nalezení koeficientu filtru se obvykle řeší optimalizace chybové funkce – ta může být stochastická nebo deterministická. Stochastický přístup požaduje mnohem větší počet měření ke kalkulaci statistiky – využíváno algoritmy typu LMS a NLMS. Deterministický přístup požaduje výpočet charakteristik z velkého počtu vzorků – využíváno algoritmy typu RLS a QR-RLS [5].

4.3.1 Algoritmus LMS

Algoritmus LMS patří mezi nejrozšířenější a nejpoužívanější adaptivní algoritmy používané v současné praxi. Síla algoritmu LMS spočívá hlavně v jednoduchosti a matematické nenáročnosti. Toto široké spektrum aplikací algoritmu LMS lze přičíst jeho jednoduchosti a robustnosti [3]. Tyto algoritmy jsou založeny na gradientním vyhledávacím algoritmu nazývaném také metoda největšího spádu. Závislost střední kvadratické odchylky výstupního chybového signálu adaptivního FIR filtru na koeficientech filtru je kvadratická křivka s jedním globálním minimem [6]. Výstupní rovnice upravíme podle rovnice (4.5), tedy:

$$y(n) = \mathbf{w}(n)\mathbf{x}(n), \quad (4.7)$$

$$y(n) = \mathbf{w}^T(n)\mathbf{x}(n). \quad (4.8)$$

Chybový signál určíme jako rozdíl požadovaného a výstupního signálu, tedy:

$$e(n) = d(n) - y(n). \quad (4.9)$$

Rekurze filtru je dána vztahem (4.10):

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e(n)\mathbf{x}(n), \quad (4.10)$$

kde μ představuje velikost kroku adaptivního filtru, $\mathbf{w}(n)$ je vektor koeficientů filtru, $\mathbf{x}(n)$ je vstupní vektor filtru. Velikost kroku filtru je zásadní parametr, který má schopnost zlepšit rychlost konvergence adaptivního filtru [8].

4.3.2 Algoritmus NLMS

Normalizovaný LMS (NLMS) algoritmus vznikl modifikací standardního LMS algoritmu. NLMS algoritmus aktualizuje koeficienty adaptivního filtru za použití rovnice (4.11):

$$\mu(n) = \frac{\mu}{\|x(n)\|^2}. \quad (4.11)$$

Je zřejmé, že NLMS algoritmus je téměř identický s jeho standardní variantou, algoritmem LMS. Rozdíl mezi těmito algoritmy je ten, že u algoritmu NLMS je při velkých vstupních hodnotách $x(n)$ postup k minimu pomalejší a naopak. To znamená, že je využita hodnota konvergenční konstanty pouze pro první iteraci. Poté je adaptivně přepočítávána.

4.3.3 Algoritmus RLS

Algoritmus RLS je základní představitel třídy rekurzivních algoritmů, které jsou založeny na teorii Kalmanovy filtrace, časovém průměrování a metodě nejmenších čtverců. Základním rozdílem oproti algoritmu LMS je vlastní statistické pojetí. Zde se pracuje s průměrnými hodnotami veličin, které jsou počítány z časových vývojů.

Struktura filtru zůstává stejná jako u LMS algoritmu, jen adaptivní proces je odlišný vzhledem k použití průměrů [7]. RLS algoritmy vynikají především vynikajícími výkony při práci v časově proměnných prostředích a extrémně rychlou konvergencí.

Nevýhodou těchto algoritmů jsou výpočetně náročné matematické operace než u algoritmů LMS. Dále RLS algoritmy mají o jeden řád vyšší složitost a jsou tedy výrazně pomalejší [8]. Výstupní rovnice RLS algoritmu je vypočítána pomocí rovnice (4.12):

$$y(n) = \mathbf{w}^T(n-1)\mathbf{x}(n). \quad (4.12)$$

Chybový signál je vyjádřen stejným vztahem jako v rovnici (4.9). Aktualizace koeficientů filtru je dána vztahem (4.13):

$$\mathbf{w}(n+1) = \mathbf{w}^T(n) + e(n)\mathbf{K}(n), \quad (4.13)$$

kde $\mathbf{w}(n)$ je vektor koeficientů filtru, $\mathbf{K}(n)$ je vektor zisků, který je definován rovnicí (4.14):

$$\mathbf{K}(n) = \frac{\mathbf{P}(n)\mathbf{u}(n)}{\lambda + \mathbf{u}^T(n)\mathbf{P}(n)\mathbf{u}(n)}, \quad (4.14)$$

kde $\mathbf{P}(n)$ je inverzní autokorelační matice. Adaptace matice se provede podle rovnice (4.15):

$$\mathbf{P}(n+1) = \lambda^{-1}\mathbf{P}(n) - \lambda^{n-1}\mathbf{K}(n)\mathbf{u}^T(n)\mathbf{P}(n). \quad (4.15)$$

4.3.4 Algoritmus QR-RLS

Algoritmus QR-RLS je odvozeninou algoritmu RLS, který k přepočtu inverzní autokorelační matice používá tzv. QR rozklad. Tento rozklad provádí zápis matice jako součin dvou matic, z nichž jedna je ortogonální, a druhá je v horním trojúhelníkovém tvaru. QR rozklad se provádí podle klasického nebo také modifikovaného Gramova-Schmidtova algoritmu.

QR rozklad zaručuje tu vlastnost, že je číselně stabilnější než standardní RLS algoritmus. Nicméně algoritmus QR-RLS vyžaduje více výpočetních zdrojů než standardní RLS algoritmus [8]. Postup výpočtu samotného signálu je obdobný jako u RLS algoritmu. Více o QR rozkladu v [14].

5 Praktická realizace vlastního systému pro adaptivní filtraci

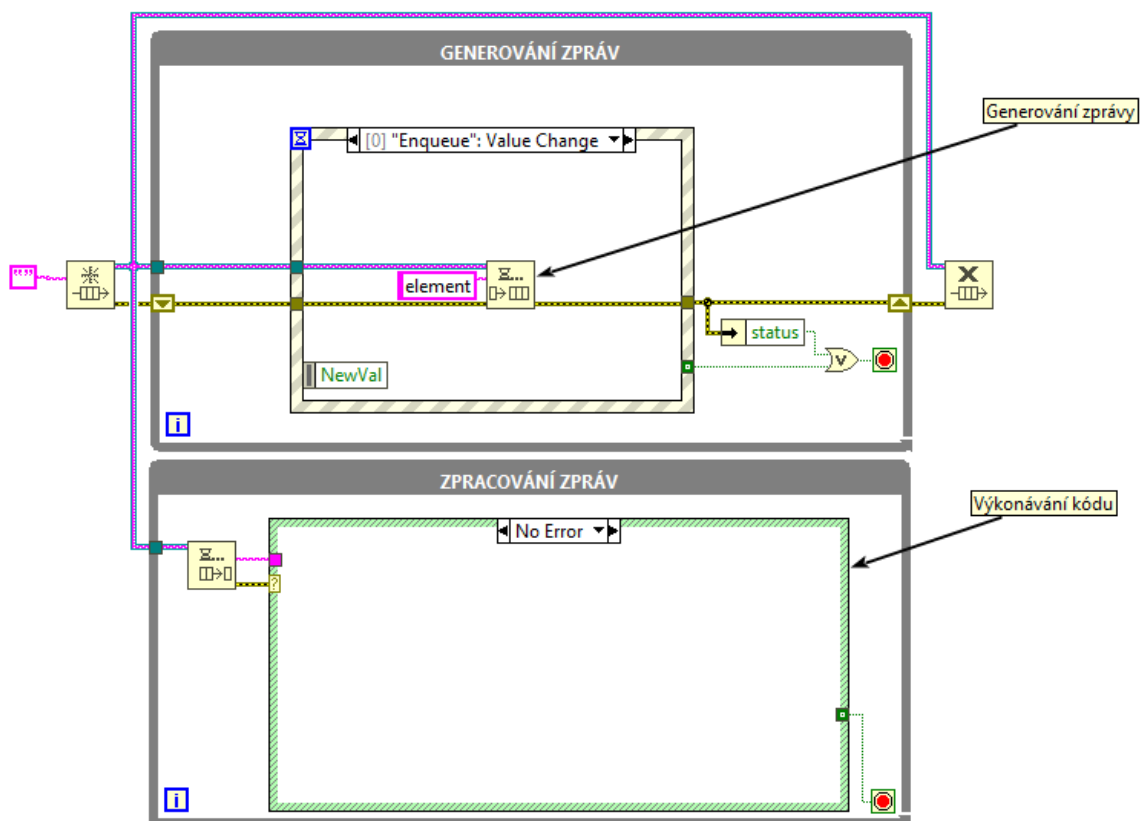
Výsledek bakalářské práce sestává ze dvou programů vytvořených v programovacím a vývojovém prostředí LabVIEW. První program představuje online aplikaci pro zpracování řečových signálů v časové a frekvenční oblasti. Program dále umožňuje adaptivního filtrování v reálném čase, výpočet globálního SNR a ukládání dat. Podrobný popis online aplikace v příloze *Technický popis online aplikace.pdf*. Nahrávky řeči a šumu byly pořízeny na notebooku pomocí měřicí karty NI 9234 a mikrofonů G.R.A.S. 40PP. Podrobnější popis těchto komponent v kapitole 3.

Druhý program tvoří offline aplikaci pro podrobnější zpracování adaptivních algoritmů. Jedná se o podobnou variantu prvního programu s rozdílným programovým uspořádáním.

5.1 Popis realizovaného programu v offline režimu

Aplikace by navržena i v offline režimu na bázi programové struktury producent–konzument (viz Obrázek 18), ve které byly pořízeny veškeré experimenty obsažené v bakalářské práci.

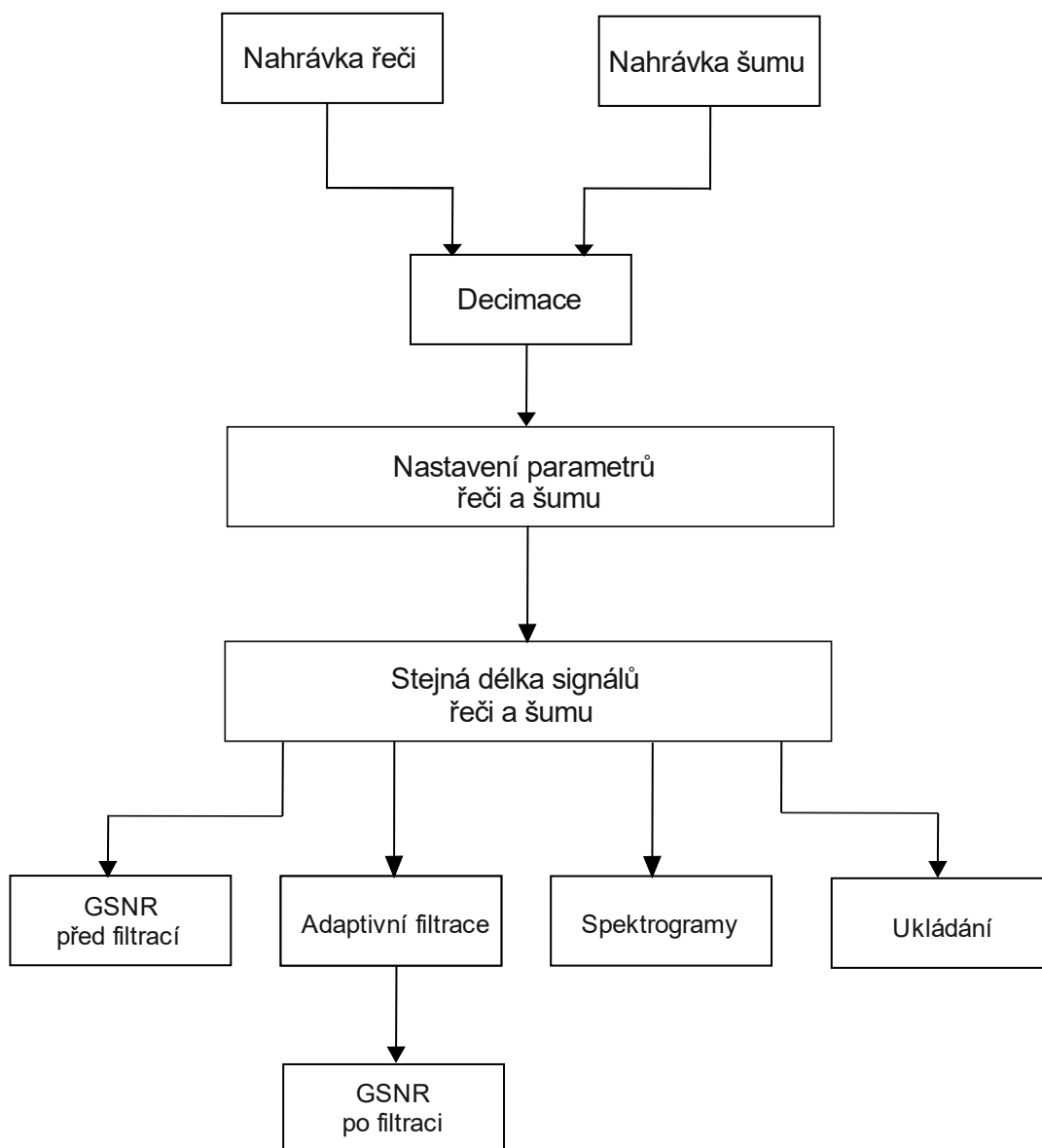
5.1.1 Architektura producent-konzument



Obrázek 18: Struktura producent-konzument.

Program je strukturován na architektuře producent-konzument. To znamená, že jedna smyčka generuje zprávy, a druhá smyčka je zpracovává. Data jsou v první smyčce umisťována do fronty, v druhé smyčce dochází k jejich vyzvedávání. Předávanými daty mezi smyčkami jsou zprávy o tom, co provedl uživatel na čelním panelu, a je tedy vykonávána vždy ta část kódu, která odpovídá náležité události způsobené uživatelem [11]. Například pokud uživatel zmáčkne tlačítko „načíst“ provede se pouze načtení nahrávek.

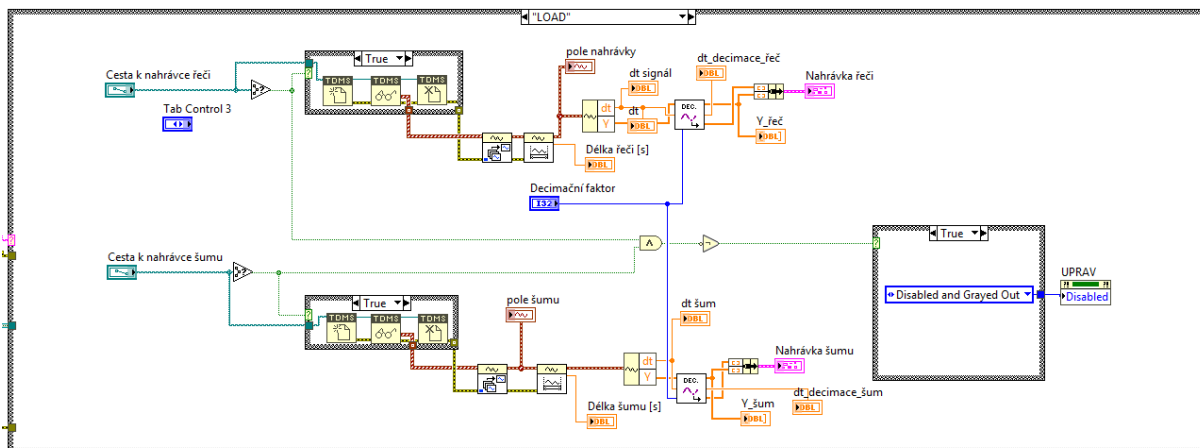
5.1.2 Algoritmus offline aplikace



Obrázek 19: Algoritmus offline aplikace.

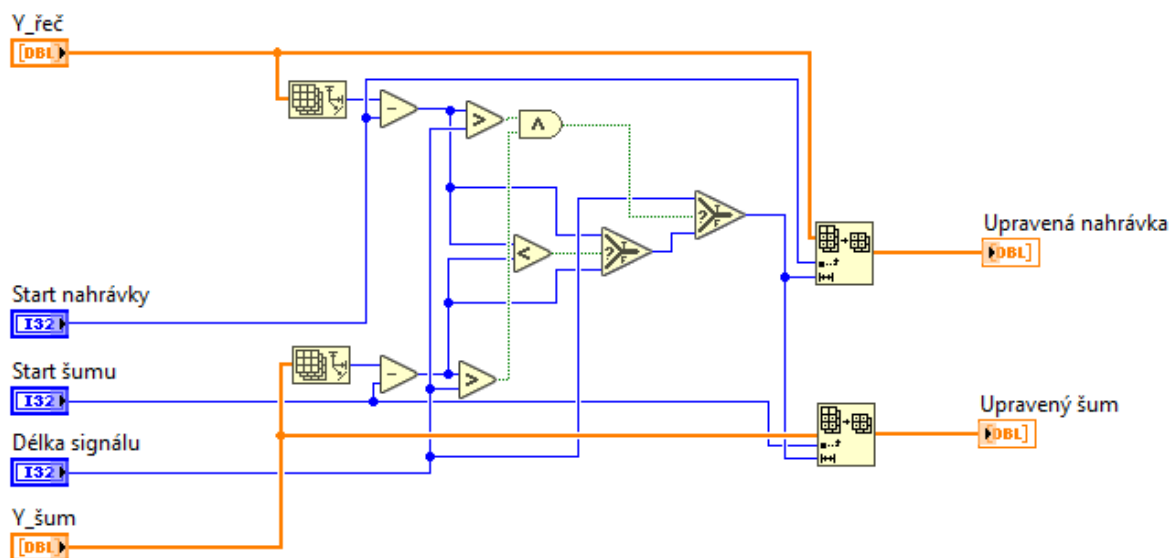
Algoritmus je sestaven tak, že je nejprve vybrána nahrávka řeči a šumu ve formátu .TDMS. Pokud jsou cesty správně zvoleny uživatel zmáčkne tlačítko „Načtení nahrávky a šumu.“ Odemkne se tlačítko pro úpravu signálů, proběhne indexace pole hodnot a zjištění délky celé nahrávky v sekundách. Nahrávky byly pořízeny na maximální vzorkovací frekvenci 51200 Hz, s kterou dokáže měřicí karta pracovat.

Proto byla vytvořena funkce pro decimaci signálů. Uživatel může volit krok decimace 1 až 6, přičemž 1 je původní nahrávka a 6 odpovídá vzorkovací frekvenci 8555 Hz z důvodu dodržení Shannon-Kotělnikovova teorému, kdy se lidská řeč pohybuje v rozsahu 300 Hz až 3,4 kHz. Po provedení decimace dojde k vykreslení nahrávek, viz Obrázek 20.



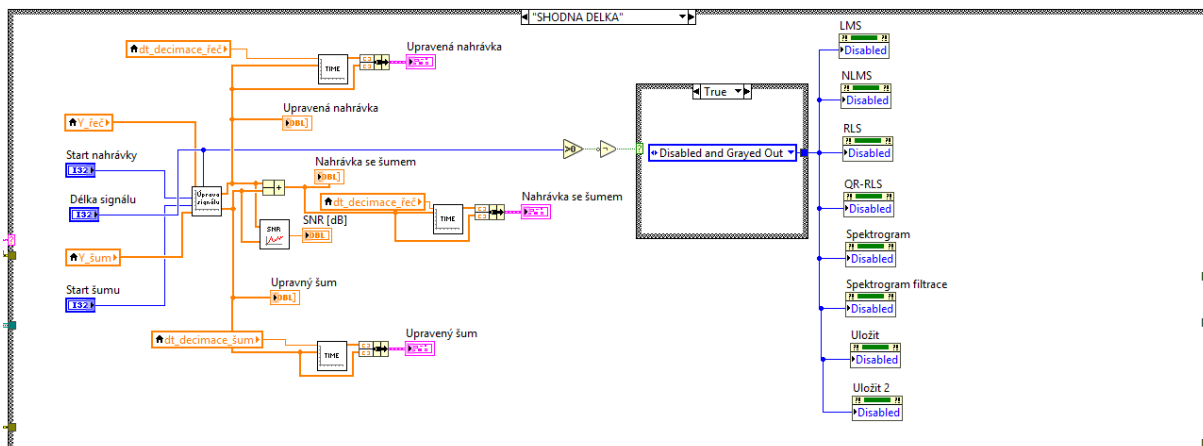
Obrázek 20: Zpracování zprávy pro načtení nahrávek.

V dalším kroku uživatel musí nastavit jakou délku bude mít celá nahrávka. Z důvodů použití adaptivní filtrace pro odfiltrování zašuměného signálu řeči, ale i kvůli korektnosti výpočtu globálního SNR je ve vývojovém prostředí LabVIEW vytvořen funkční blok *úprava signálu_sub*, který slouží proto, aby délka dvou zpracovávaných nahrávek, řeči a šumu, měla stejnou délku při pozdějších výpočtech. Jelikož je v programu možnost zvolit si délku signálu a počátek signálu, a všechny nahrávky nemají stejnou délku, tak se tento blok stará o to, aby konečná délka byla dlouhá maximálně jako ta kratší z vybraných. Řešení této funkce lze vidět na (Obrázek 21).



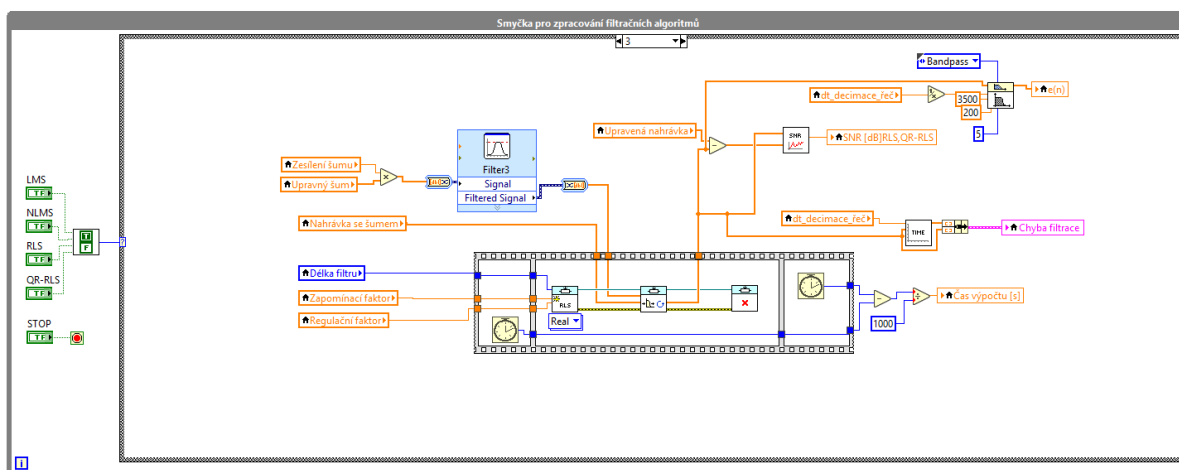
Obrázek 21: Blokový diagram vyhodnocení pro stejnou délku signálů.

Pokud je zadána délka signálu, sečtou se nahrávky řeči a šumu a dojde k odemčení tlačítek pro aktivaci adaptivních filtrů, spektrogramů a ukládání. Současně jsou upravené nahrávky přepočítány pro zobrazování času na x-ové ose a vykresleny do příslušných grafů, viz Obrázek 22.



Obrázek 22: Zpracování zprávy pro stejnou délku signálů.

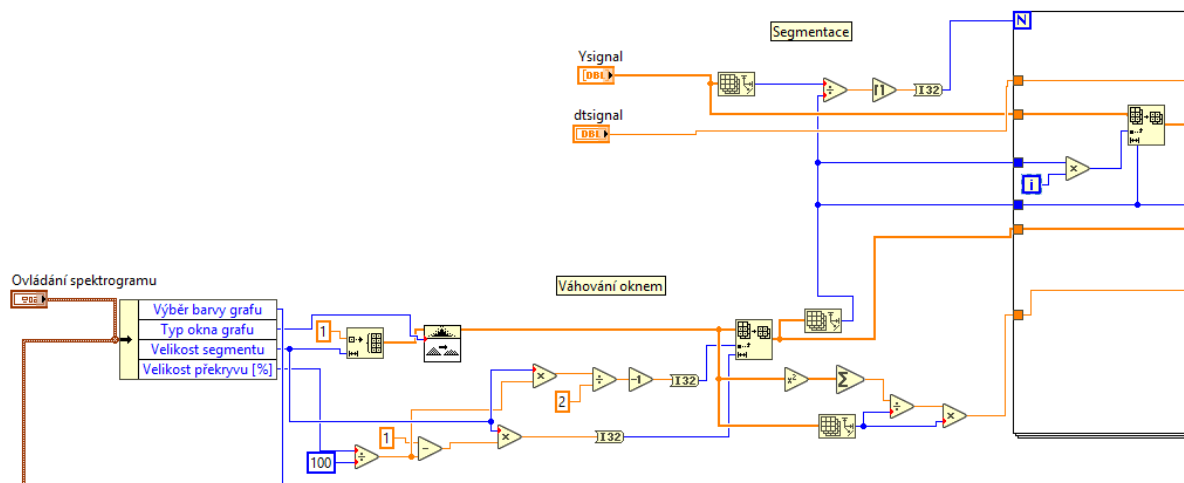
V tento moment uživatel může zapnout požadovaný adaptivní algoritmus. Aby bylo docíleno změny parametrů v reálném čase, tedy aby uživatel mohl libovolně měnit parametry algoritmů a viděl okamžité změny v grafech, byly funkce pro adaptivní algoritmy naprogramovány zvlášť mimo architekturu producent-konzument. Na (Obrázek 23) lze vidět případ zvoleného adaptivního algoritmu RLS. V místnosti při procesu mluvení dochází k odražení zvuku, který vytváří dozvuk. Z toho důvodu byla použita expresní funkce filtru typu FIR dolní propust s délkou filtru 5 a dolní mezní frekvencí 0,1 pro zpoždění šumu.



Obrázek 23: Ukázka filtrace algoritmu RLS.

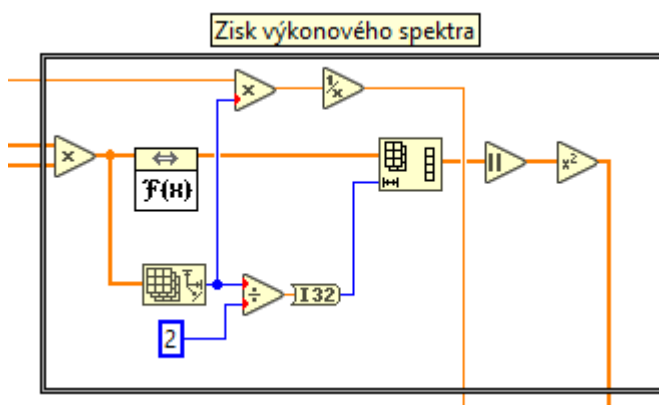
Uživatel má také možnost vykreslit spektrogramy. Pro vykreslování spektrogramu musíme nejprve zadat velikost segmentu, kterou vytvoříme pomocí funkce *initialize array* o požadovaném počtu N . Okna jsou pomocí funkce *array subset* upravena tak, aby odpovídala zadanému překryvu. K rozdělení vstupního signálu na segmenty je využita smyčka *For*. Blokem *array size* zjistíme počet vzorků N , kterou podělíme počtem N okna. Tato hodnota je zaokrouhlena k nejbližšímu celému číslu a určuje výsledný počet segmentů k rozdělení vstupního signálu. Iterace jsou ve smyčce *For* vynásobeny

s délkou okna a přivedeny na vstup bloku *array subset*. Takto určíme první prvek pro každý segment, viz Obrázek 24.

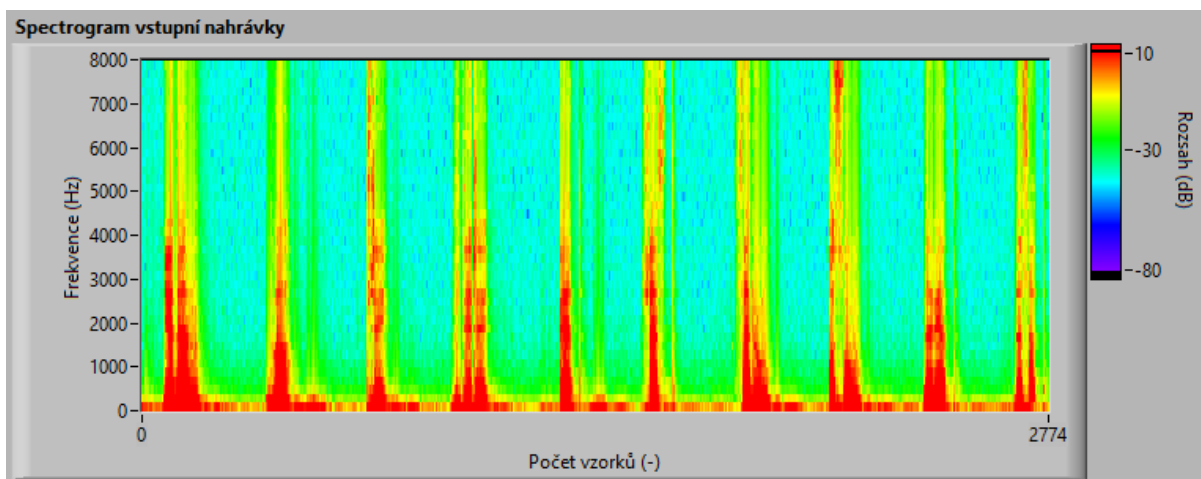


Obrázek 24: Váhování a segmentace spektrogramu.

Takto upravené hodnoty o počtu N se vynásobí oknem o stejném počtu N , a spočítáme jejich diskretní transformaci pomocí algoritmu FFT. V bloku *array subset* omezíme dvoustranné spektrum na jednostranné. Z takto upraveného spektra získáme absolutní hodnoty a spočítáme jeho druhou mocninu. Tak získáme výsledné výkonové spektrum, viz Obrázek 25. Ukázka výsledného spektrogramu je zobrazena na (Obrázek 26).

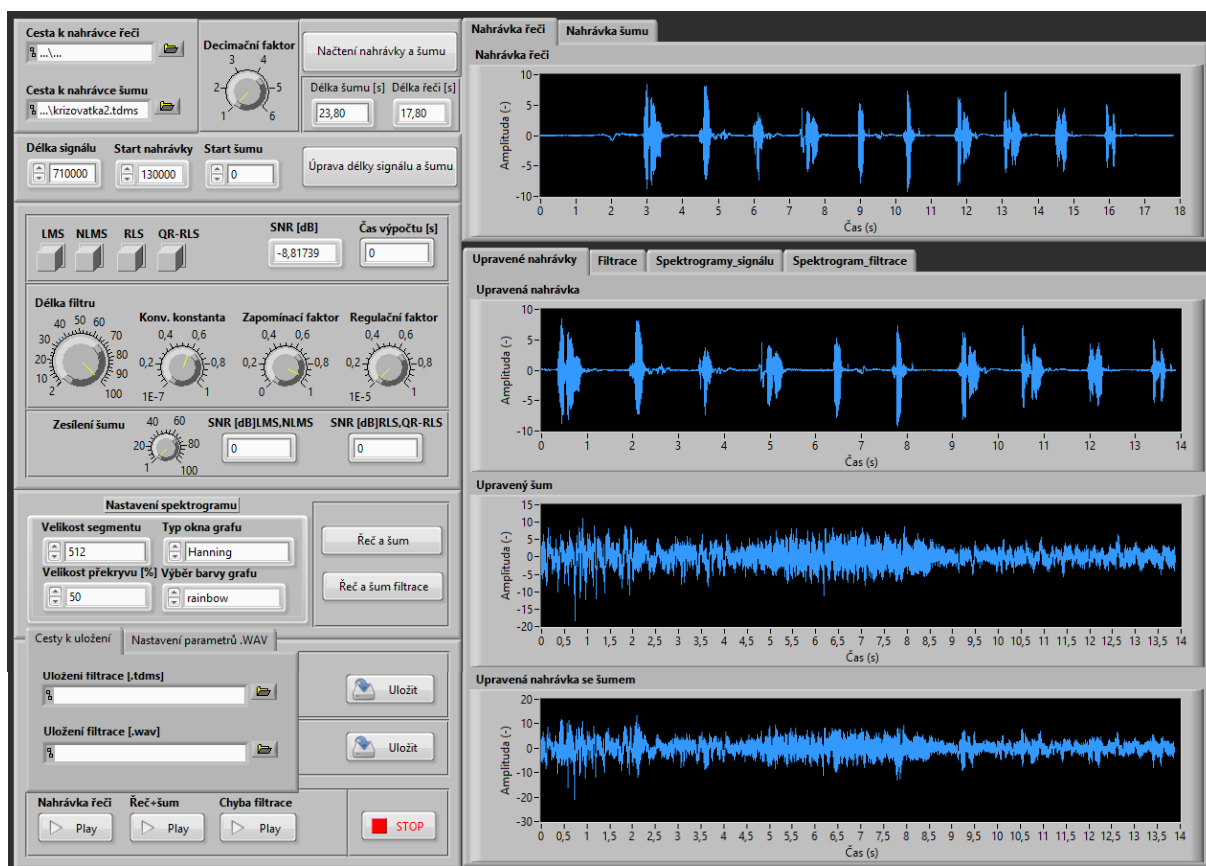


Obrázek 25: Zisk výkonového spektra.



Obrázek 26: Ukázka vykreslení spektrogramu v offline aplikaci.

Uživatel také může zesílit načtenou nahrávku šumu pro lepší analýzu adaptivních algoritmů. Dále lze zvolit ukládání filtrovaného signálu do formátu .TDMS nebo .WAV. Pro porovnání kvality filtrovaných nahrávek má uživatel možnost si přehrát nahrávku řeči, zašuměnou řeč a také odfiltrovanou řeč. Vzhled čelního panelu je zobrazen na (Obrázek 27).

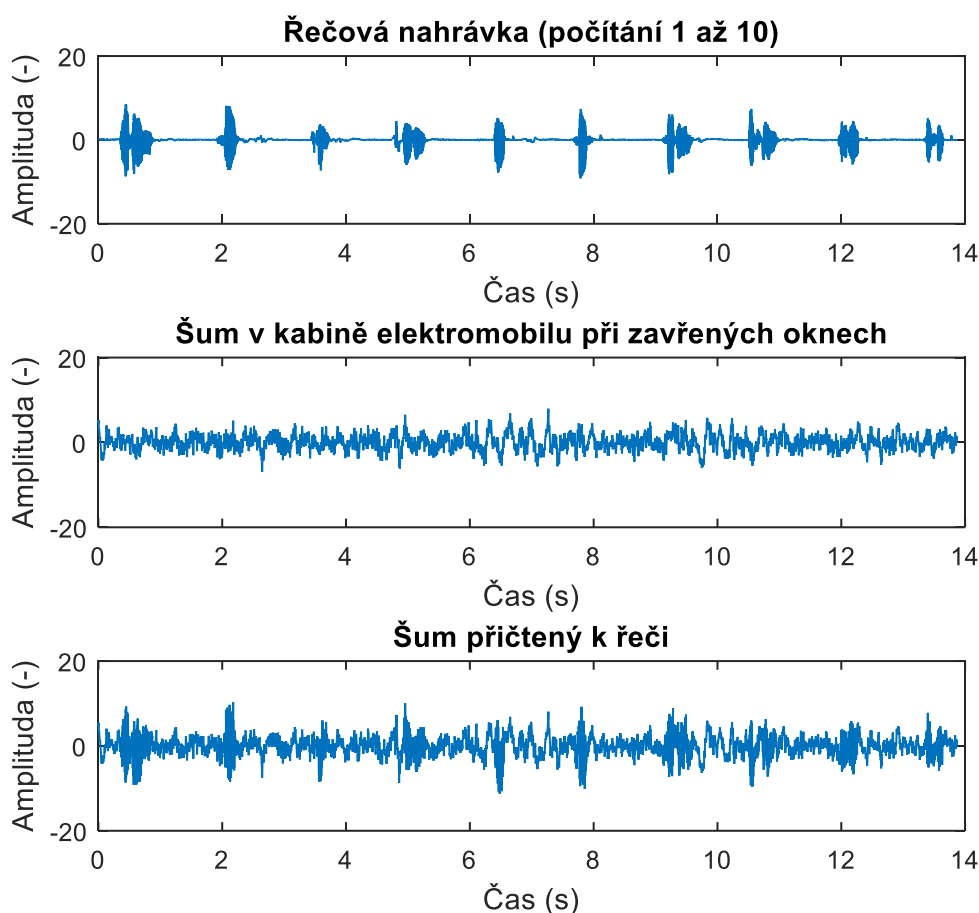


Obrázek 27: Čelní panel offline aplikace.

5.2 Změna vlastností filtru vlivem změny parametrů

V této kapitole budeme sledovat vliv změny délky filtru M a změny konvergenční konstanty μ , která má vliv na rychlost konvergence.

Pro experimenty byla zvolena řečová nahrávka o délce 14 sekund (716 800 vzorků). Přičtený šum k nahrávce řeči je naměřený hluk v kabině elektromobilu při rychlosti 50 km/h. Délka nahrávky je stejná jako u řeči, tedy 14 sekund (716 800), viz Obrázek 28. Vzorkovací frekvence obou nahrávek je 51 200 Hz.

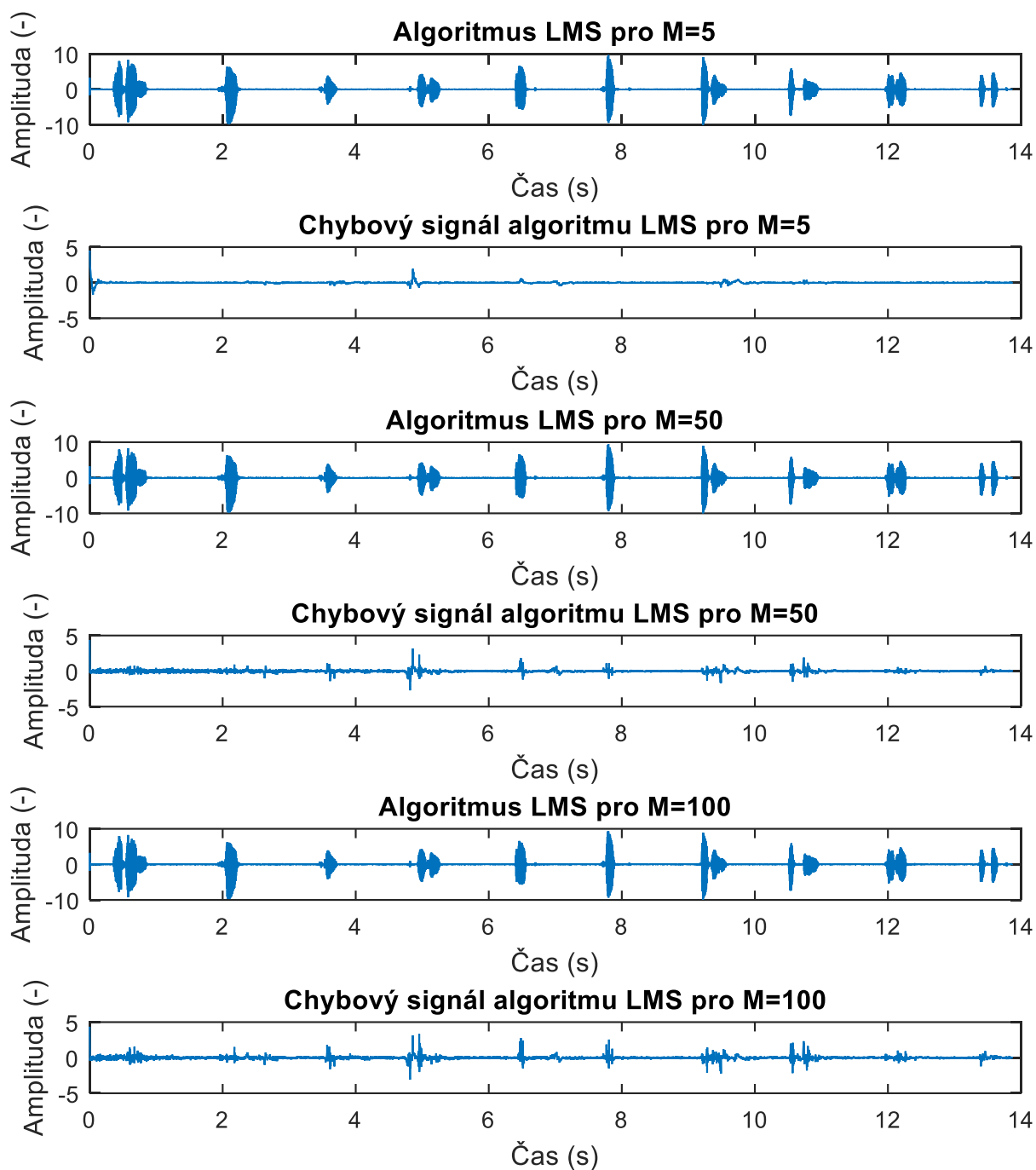


Obrázek 28: Průběh řečové nahrávky, šumu a šum přičtený k nahrávce.

5.2.1 Vliv délky filtru

Jedním z důležitých vstupních parametrů adaptivních algoritmů je délka filtru. V praxi se též uvádí řád filtru, který udává počet koeficientů filtru. Délka filtru je M a řád filtru N , přičemž platí, že $N = M - 1$. Pro algoritmus LMS byla hodnota konvergenční konstanty zvolena na hodnotu 0,1 a postupně byla zvyšována hodnota délky filtru. Z (Obrázek 29) lze vyvodit, že při zvyšování délky filtru docházelo k lepší rekonstrukci původního signálu. Výpočetní čas je u algoritmu LMS zanedbatelný (viz Tabulka 3) díky jeho matematické jednoduchosti a nenáročnosti. Dále bylo zjištěno, že i když je původní signál

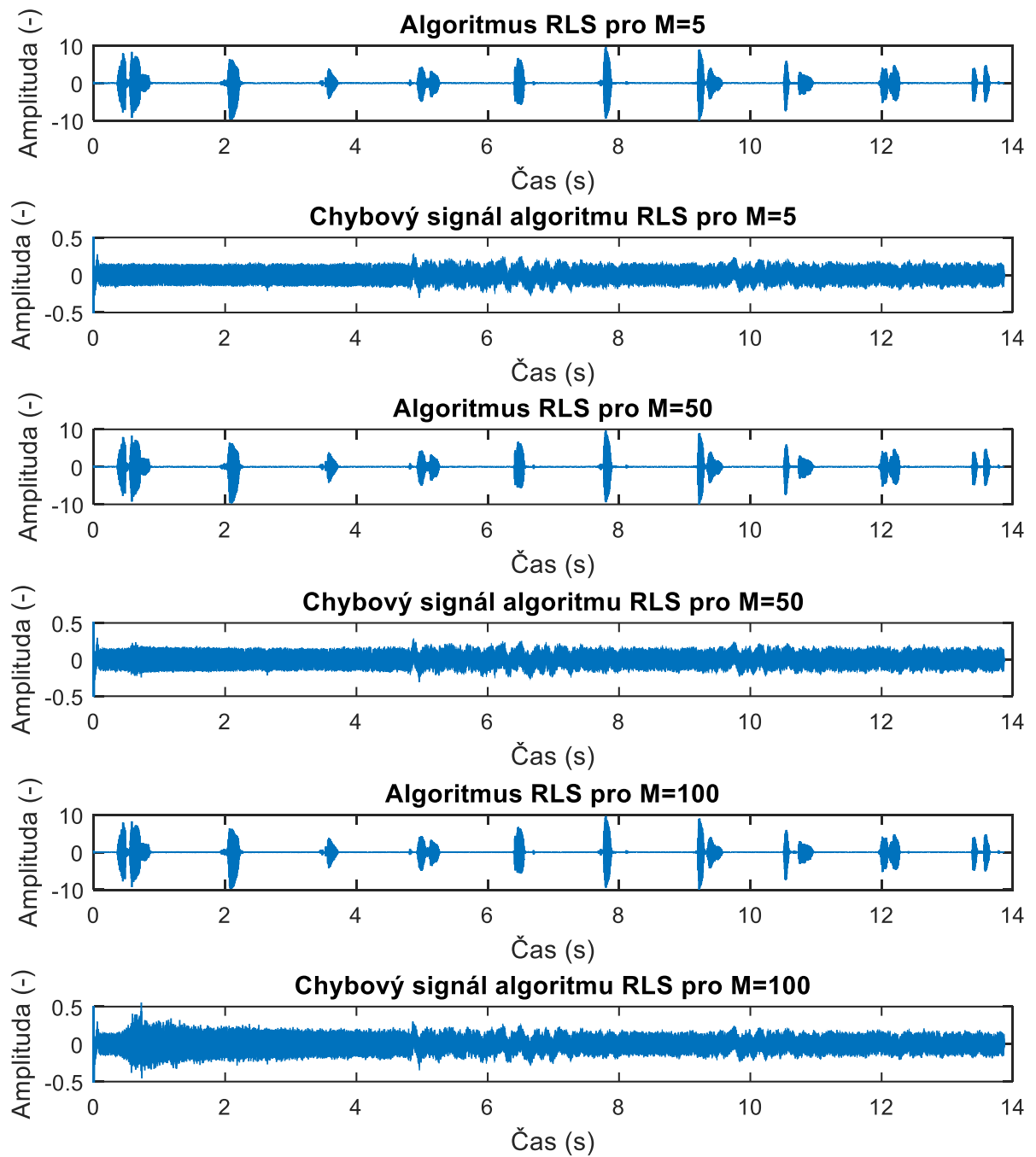
lépe rekonstruován zvedá se výsledná chyba filtrace. To může znamenat, že filtr potlačuje šum za cenu deformace užitečného signálu.



Obrázek 29: Průběhy délky filtru a chyby filtru pro algoritmus LMS.

Tabulka 3: Tabulka doby výpočtu pro algoritmus LMS.

Délka filtru M (-)	Doba výpočtu (s)
5	0,17
50	0,43
100	0,63



Obrázek 30: Průběhy délky filtru a chyby filtru pro algoritmus RLS.

U algoritmu RLS se nastavují parametry regularizačního faktoru a faktoru zapomínání λ . Z technické praxe je doporučeno nastavovat hodnotu faktoru zapomínání v rozmezí 0,98 až 1, kdy 1 značí nekonečnou paměť. Pro testování délky filtru byl nastaven regularizační faktor na hodnotu 0,00001. Tato hodnota je implicitně nastavena ve vývojovém prostředí LabVIEW a faktor zapomínání $\lambda = 1$. Z (Obrázek 30) je patrné, že při zvyšování délky filtru podobně, jako u algoritmu LMS dochází k lepší rekonstrukci původního signálu. Rozkmity chybových signálů byly ± 20 . Z důvodu kvalitnějšího vykreslení byly hodnoty nastaveny na hodnotu $\pm 0,5$. Dále bylo zjištěno, že při zvyšování délky filtru dochází k deformaci/zkreslení signálů způsobením velkého rozkmitu na vstupu. Ten je zapříčiněn právě velikostí délky filtru. Čím bude délka filtru vyšší, tím bude větší rozkmit.

Oproti algoritmu LMS se algoritmus RLS dokáže mnohem rychleji přiblížit k optimální hodnotě filtrace na úkor vyšší výpočetní náročnosti, viz Tabulka 4.

Tabulka 4: Délka výpočtu pro algoritmus RLS.

Délka filtru M (-)	Doba výpočtu (s)
5	0,37
50	14,88
100	61,63

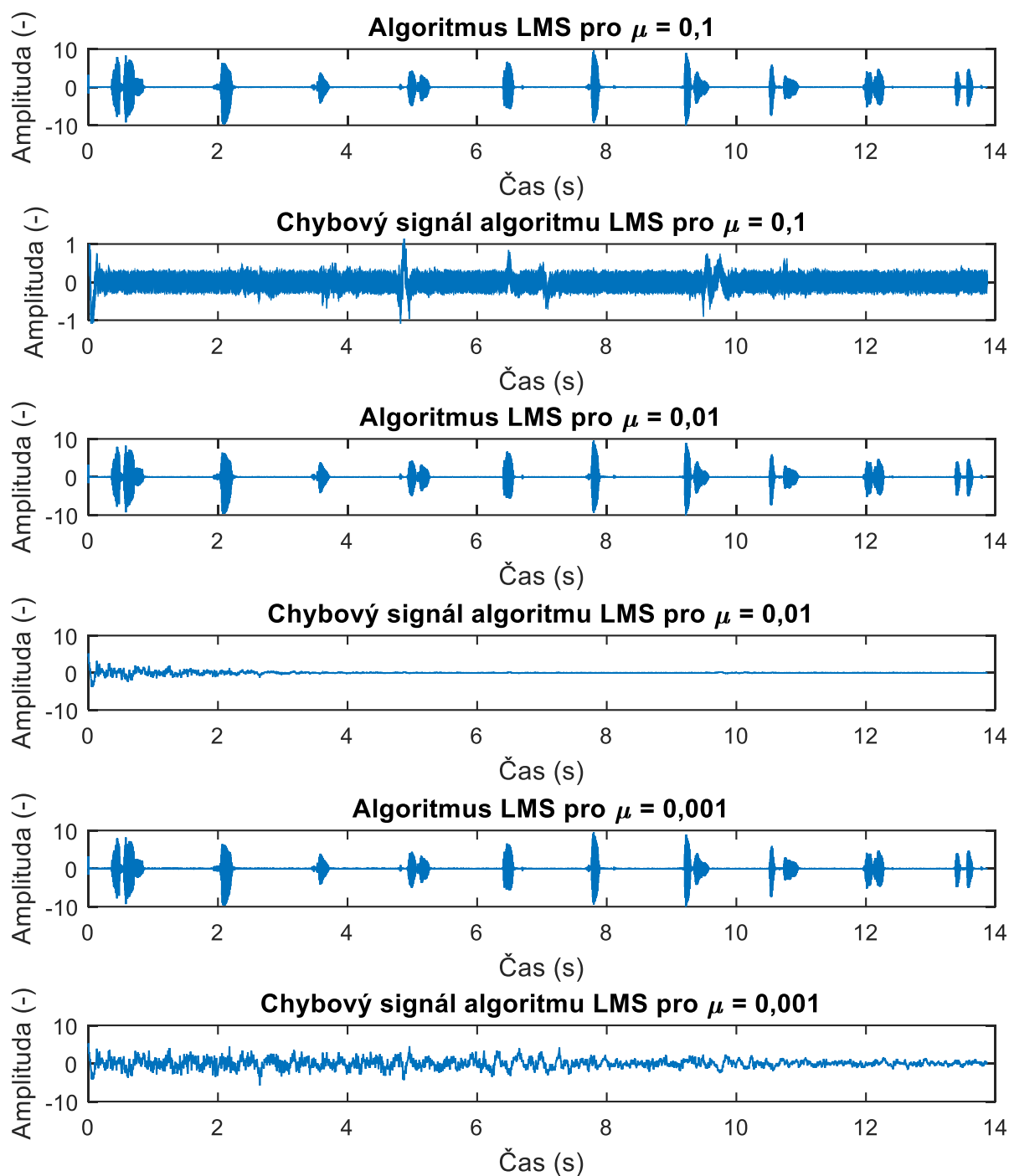
5.2.2 Vliv konvergenční konstanty

Při posuzování vlivu délky konvergenční konstanty μ bylo zjištěno, že má vliv na rychlost adaptace vah. Jde o malou kladnou konstantu, která ovlivňuje vlastnosti adaptace algoritmu. Je-li velká, dochází díky přítomnosti užitečného signálu k rozvažování vah. Pojem malé a velké hodnoty parametru μ souvisí s rozptylem vstupního signálu.

Pro vyřešení dilematu pomalá konvergence vs. rozvažování vah lze použít například detektor řeči (VAD). Ten zastaví adaptaci vah při přítomnosti řečového signálu. K adaptaci pak dochází pouze v pauzách mezi řečí.

Z (Obrázek 31) lze vidět, že při snižování hodnoty konvergenční konstanty bude algoritmu trvat déle, než se ustálí na optimální hodnotě. To znamená, že algoritmus potřebuje větší počet iterací, aby se adaptivní filtr přiblížil k optimálnímu stavu filtrace. Dále při nastavené hodnotě konvergenční konstanty na hodnotu 0,001 je patrné, že filtru trvá příliš dlouhou dobu, než se adaptuje na optimální hodnotu.

Jelikož je algoritmus NLMS téměř identický s jeho standardní variantou, algoritmem LMS bude počet iterací obou algoritmů shodný. Jediný rozdíl mezi těmito algoritmy je, že hodnota konvergenční konstanty je využita pouze pro první iteraci. Poté je adaptivně přepočítávána. Z těchto důvodů byla ukázána pouze konvergenční konstanta pro algoritmus LMS.



Obrázek 31: Průběhy konvergenční konstanty pro algoritmus LMS.

Pro shrnutí vlastností filtrů byly použity chybové signály délek filtru a chybové signály konvergenčních konstant algoritmu LMS. Z těchto signálů byla zjištěna doba ustálení a odečtené maximální absolutní chyby. V praxi se obecně považuje za vhodné co nejmenší řád filtru, a naopak co největší konvergenční konstanta, aby bylo docíleno rychlé adaptaci filtru, tedy minimálního výpočetního nároku a co nejefektivnější filtrace. Z (Tabulka 5) a (Tabulka 6) vyplývá, že požadavky na nízkou chybu filtrace a co nejrychlejší adaptaci systému jsou vzájemně protichůdné. Systém rychle se adaptující vykazuje vysokou chybu filtrace a naopak. Vlastnosti filtru můžeme ovlivnit vhodnou volbou konvergenční

konstanty a délky filtru. Čím vyšší bude konvergenční konstanta μ , tím se bude systém rychleji přizpůsobovat za cenu vyšší chyby filtrace. Podobně pro délku filtru platí, že čím bude delší, tím kratší doba ustálení a vyšší chyba.

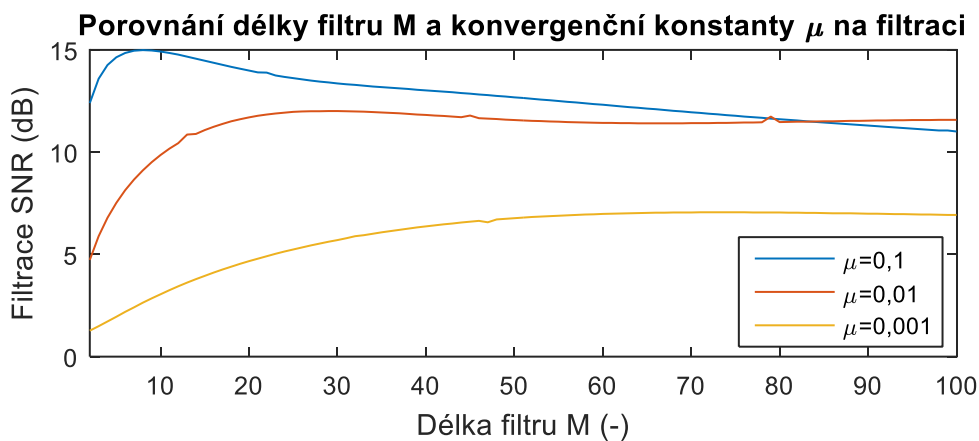
Tabulka 5: Maximální chyba a doba ustálení pro různou délku filtru.

M (-)	5	50	100
Max. absolutní chyba (-)	2,76	5,84	6,42
Doba ustálení (s)	0,33	0,19	0,02
Počet iterací (vzorky)	16 896	9 728	1024

Tabulka 6: Maximální chyba a doba ustálení pro různou délku konvergenční konstanty.

μ (-)	0,1	0,01	0,001
Max. absolutní chyba (-)	2,57	0,46	Příliš dlouhá
Doba ustálení (s)	0,27	5,42	Příliš dlouhá
Počet iterací (vzorky)	13 824	277 504	Příliš dlouhá

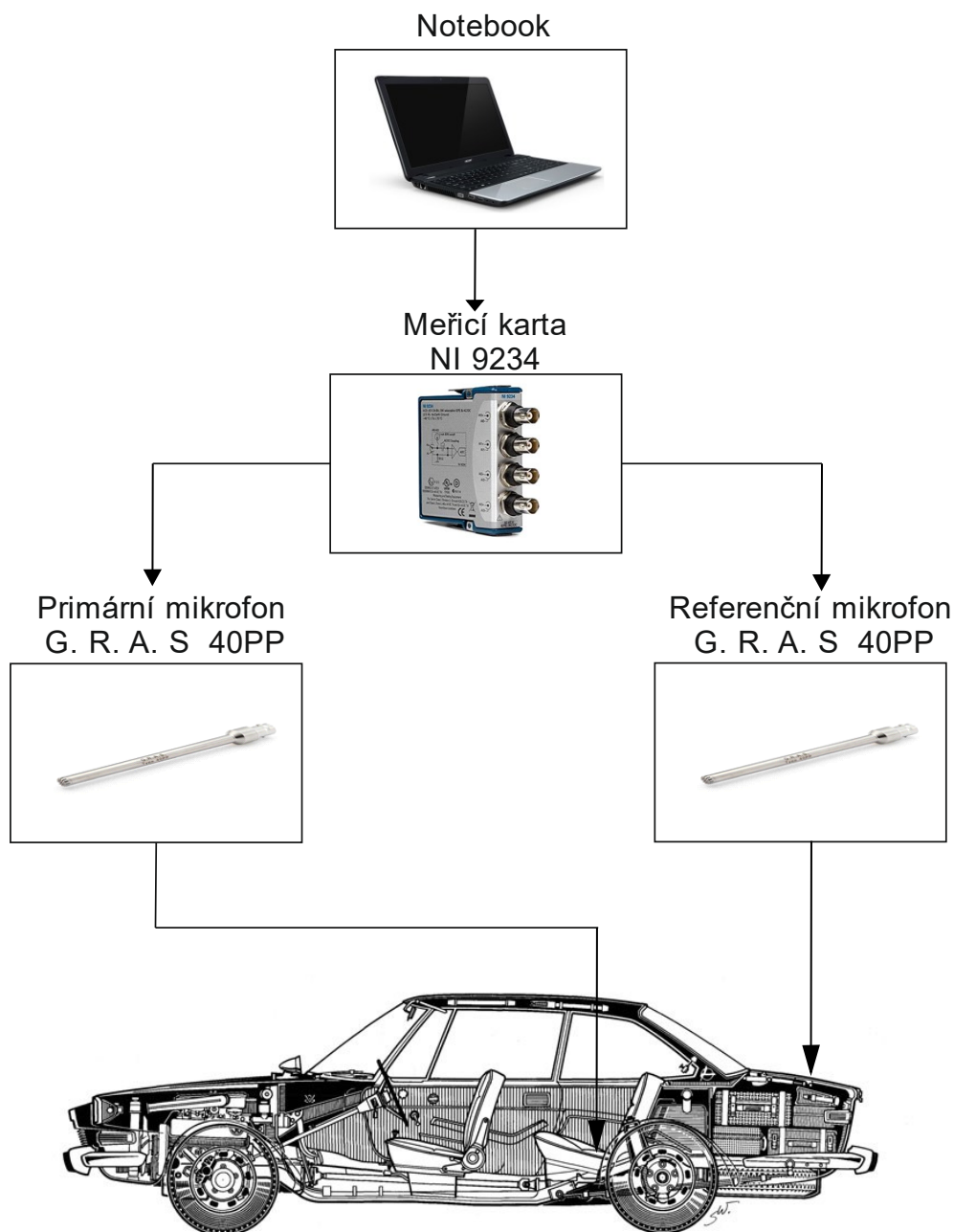
Pro ilustraci bylo proveden závislosti délky filtru a konvergenční konstanty na filtraci. Pro hodnotu konvergenční konstanty $\mu = 0,1$ je dobře vidět, že tento algoritmus konverguje pomalu. Pokud bychom si přešli tento odfiltrovaný signál, slyšeli bychom, jak postupně klesá hladina hluku. Jak už bylo řečeno algoritmus tedy potřebuje větší počet iterací, aby se adaptivní filtr přiblížil k optimálnímu stavu filtrace. Dále bylo zjištěno, že hodnotu $\mu = 0,001$ je příliš malá, aby se filtr dokázal optimálně adaptovat.



Obrázek 32: Vliv délky filtru a konvergenční konstanty na filtraci.

5.3 Efektivita filtrace adaptivních algoritmů

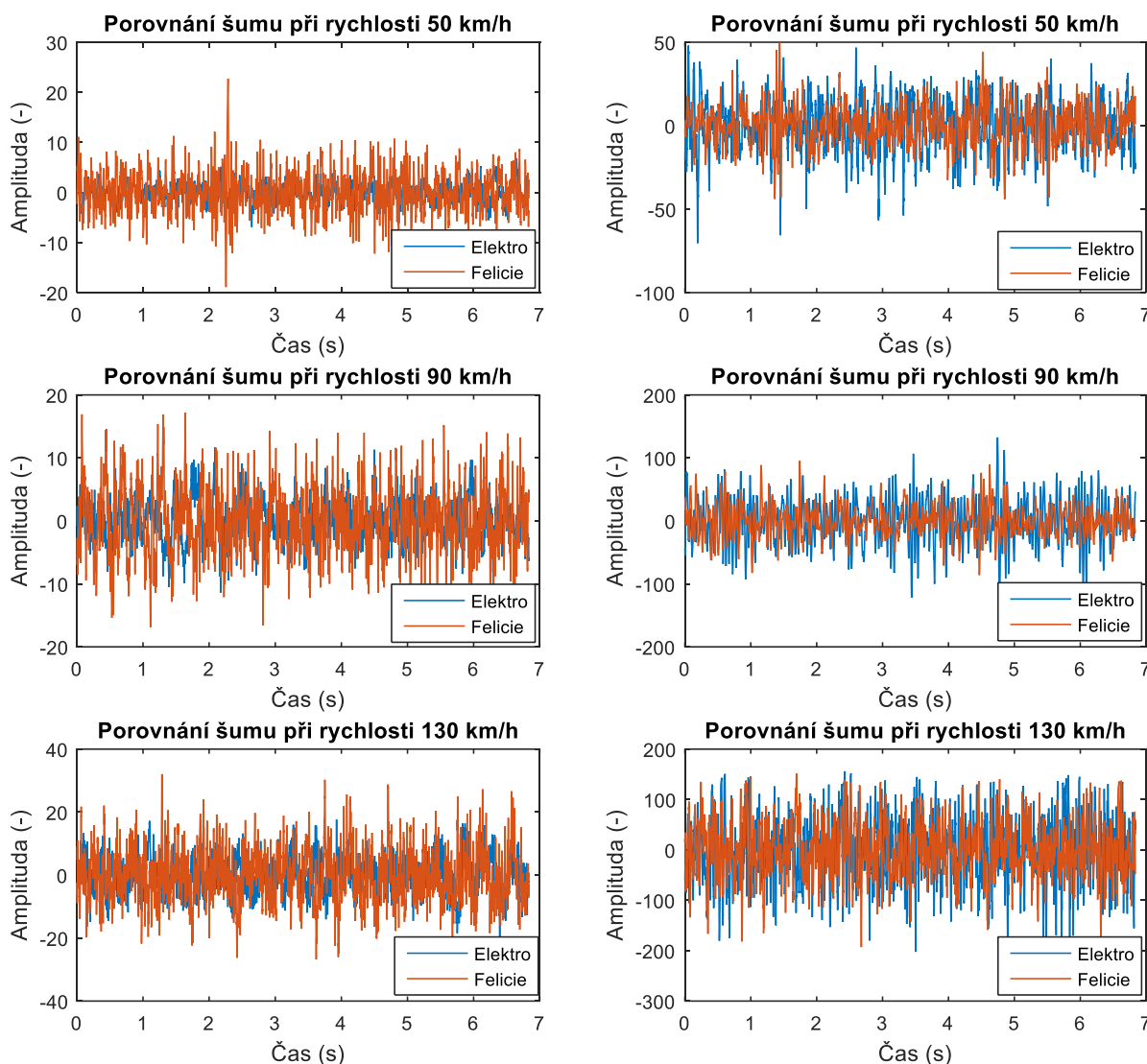
U komerčních aplikací z oblasti sdělovací techniky jako jsou např. telekonference, audio akustika apod. se často setkáváme se situacemi, ve kterých musíme čelit problémům se šumem. Pro testování míry úspěšnosti filtrace byla provedena měření rušení v elektromobilu a ve vozidle Škoda Felicia. Měření rušení proběhlo při rychlostech 50, 90 a 130 km/h v centru Ostravy a na dálnici D1. Rušení bylo měřeno na zadním sedadle a v kufru automobilu nejprve při zavřených oknech a následně při otevřených, viz Obrázek 33. Vzorkovací frekvence byla 51 200 Hz.



Obrázek 33: Blokové schéma měření rušení.

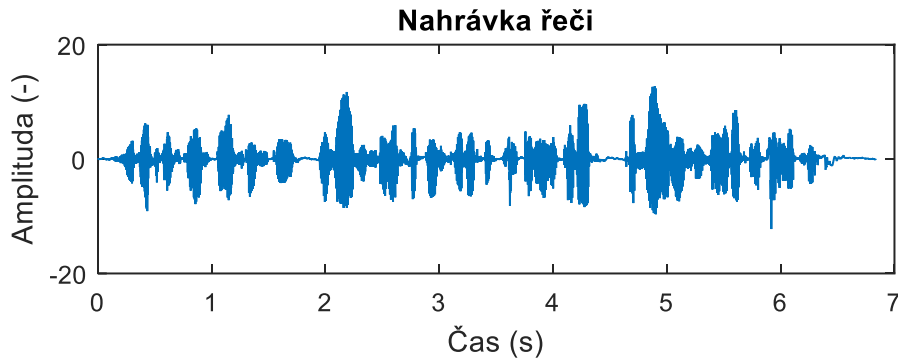
Nahrávky rušení byly vytvořeny pro testování adaptivních algoritmů, které dokáží eliminovat šum vzdáleného mluvčího, který se dostává zpět akustickou vazbou stejně jako řeč. Například při jízdě automobilu, kdy řečník komunikuje pomocí hands-free soupravy mobilního telefonu, ve které jsou zabudovány právě systémy adaptivních algoritmů pro potlačení šumu.

Ukázka naměřených rušení je znázorněna na (Obrázek 34). Z průběhu lze vyčíst, že Škoda Felicie vytváří oproti elektromobilu při zavřených oknech větší hluk, avšak při otevřených oknech je tento jev téměř nepodstatný. To je převážně způsobeno například různou intenzitou větru nebo dalšími vnějšími vlivy.



Obrázek 34: Průběhy rušení–primární mikrofon zavřená okna (vlevo), otevřená okna (vpravo).

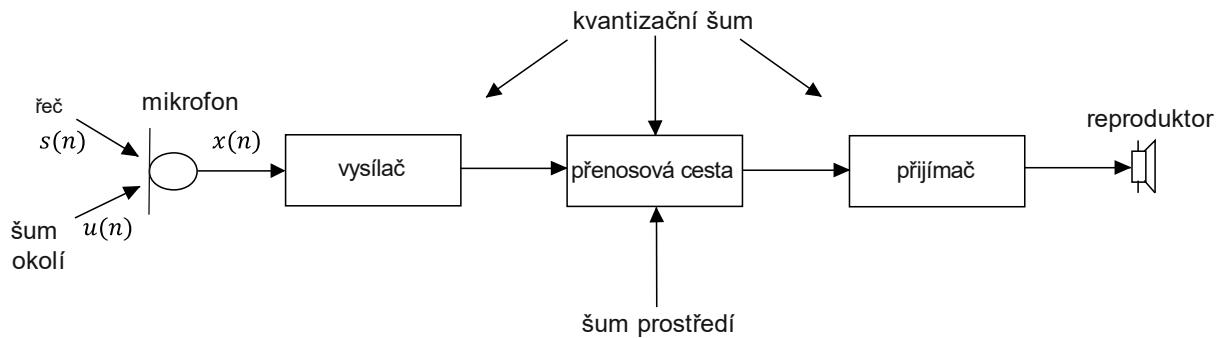
K těmto rušením byla přičtena řečová nahrávka o délce 7 sekund (350000 vzorků), která představuje větu: „Život v lidské společnosti je závislý na schopnosti jednotlivců komunikovat a vzájemně si sdělovat informace.“ Vzorkovací frekvence nahrávky je též 51 200 Hz.



Obrázek 35: Nahrávka řeči.

5.3.1 Objektivní kritérium posouzení kvality filtračního procesu

Při analýze řečového signálu se nejčastěji setkáme s dvěma základními možnými rušeními: aditivní šum a konvoluční zkreslení. Aditivní šum může být výsledkem operací v pevné řadové čárce při kódování a zpracování přenášeného signálu, tzv. kvantizační šum, nebo může být přičten k řeči jako šumové pozadí prostředí, ve kterém je řeč snímána, popřípadě může představovat šum přenosové cesty. Naproti tomu konvoluční šum bývá způsoben přeslechy nebo změnami parametrů přenosového prostředí [9]. Schematický výskyt těchto šumů je znázorněn na (Obrázek 36).



Obrázek 36: Výskyt šumu při přenosu řeči.

Standardním kritériem pro měření úrovně šumu v signálu je *odstup signálu od šumu* – SNR. V důsledku silné nestacionarity řečového signálu však není prostá aplikace tohoto kritéria na řeč optimální a v různých parametrech je možné setkat se s různými definicemi SNR pro účely měření šumu v řeči [9]. Podrobný popis těchto kritérií lze nalézt v [10].

Je-li $s(n)$ řečový signál s výkonem σ_s^2 a $n(n)$ aditivní šum s výkonem σ_n^2 představující rušivé pozadí, lze pro jejich směs $x(n) = s(n) + n(n)$ mající výkon σ_x^2 definovat kritérium (5.1) [9]:

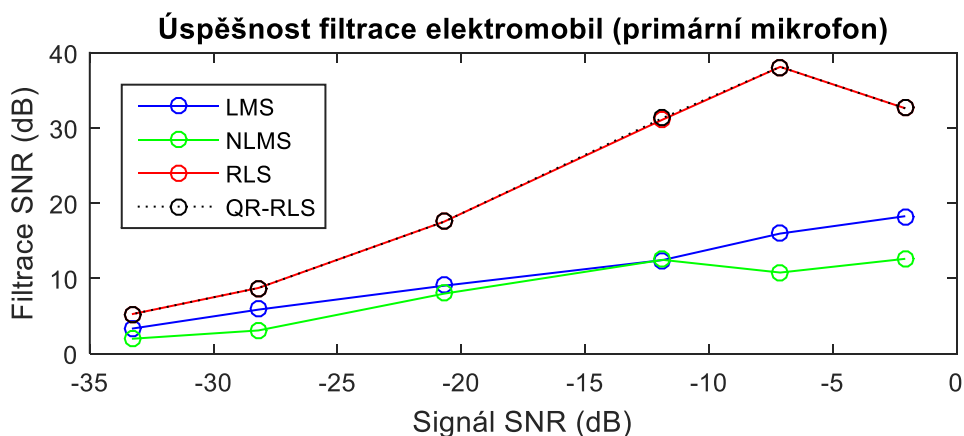
$$GSNR = 10 \log \frac{\sigma_s^2}{\sigma_n^2} = 10 \log \frac{\sum_{n=0}^{l-1} s^2(n)}{\sum_{n=0}^{l-1} n^2(n)}, \quad (5.1)$$

kde l je délka signálu.

Pro testování míry úspěšnosti adaptivních algoritmů byla použita metoda globálního SNR. Tato subjektivní metoda hodnocení kvality řeči slouží k výpočtu odstupu signálu od šumu přes celý řečový signál, aniž by se v něm nacházela řeč. Proto je nutné nastavit délku signálu tak, aby na začátku a konci nahrávky byly co nejmenší mezery, které by vedly k horšímu výsledku SNR. Pro testování byly zvoleny nahrávky rušení primárního mikrofону, tedy toho, do kterého mluvíme. Z výsledků vyplývá, že nejlépe si vedly algoritmy RLS a QR-RLS, které vykazovaly totožnou hodnotu filtrace. Dále bylo zjištěno, že NLMS algoritmus vykazoval o něco horší filtrace než jeho standardní varianta algoritmu LMS. To může být způsobeno velikostí vstupního signálu, kdy jsou hodnoty brány pouze pro první iteraci a následně jsou adaptivně přepočítány. Je také patrné, že adaptivní algoritmy mají práh, do kterého jsou schopny filtrovat. Z (Obrázek 37) lze vidět, že čím bude výkon rušení vyšší, tím se bude výsledná filtrace blížit nule. Pokud bychom takto postupovali dále dostali bychom se do záporných hodnot výsledné filtrace. To znamená, že filtr je v tomto stavu nestabilní a nedokáže úspěšně filtrovat.

Tabulka 7: Výsledky pro jednotlivé filtrace, elektromobil.

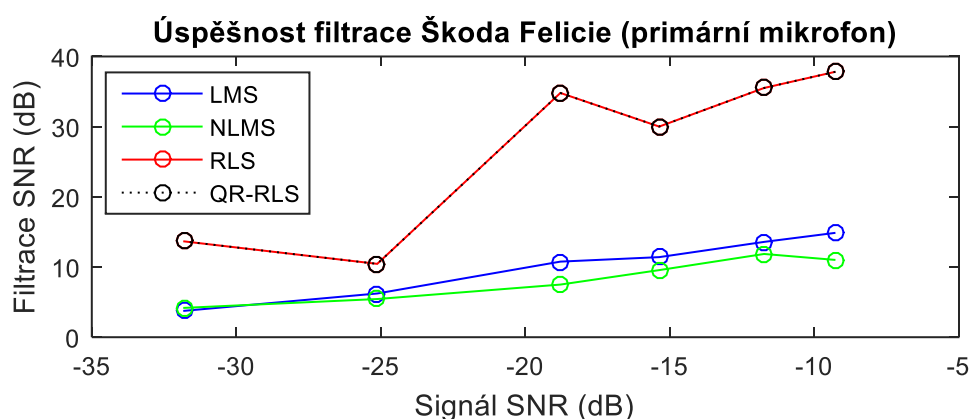
Typ šumu	SNR (dB)				
	Signál	LMS	NLMS	RLS	QR-RLS
Zavřená okna 50 km/h	-2.07	18.27	12.60	32.62	32.62
Zavřená okna 90 km/h	-7.12	15.96	10.77	38.13	38.13
Zavřená okna 130 km/h	-11.87	12.43	12.46	31.11	31.32
Otevřená okna 50 km/h	-20.70	9.02	7.96	17.53	17.52
Otevřená okna 90 km/h	-28.19	5.84	3.07	8.72	8.72
Otevřená okna 130 km/h	-33.26	3.33	1.97	5.26	5.26



Obrázek 37: Graf úspěšnosti filtrace, elektromobil.

Tabulka 8: Výsledky pro jednotlivé filtrace, Škoda Felicie.

Typ šumu	SNR (dB)				
	Signál	LMS	NLMS	RLS	QR-RLS
Zavřená okna 50 km/h	-9.26	14.87	11.01	37.82	37.82
Zavřená okna 90 km/h	-11.73	13.58	11.86	35.51	35.51
Zavřená okna 130 km/h	-15.36	11.41	9.56	30.01	30.01
Otevřená okna 50 km/h	-18.78	10.77	7.50	34.80	34.80
Otevřená okna 90 km/h	-25.12	6.23	5.44	10.45	10.45
Otevřená okna 130 km/h	-31.80	3.76	4.18	13.65	13.65

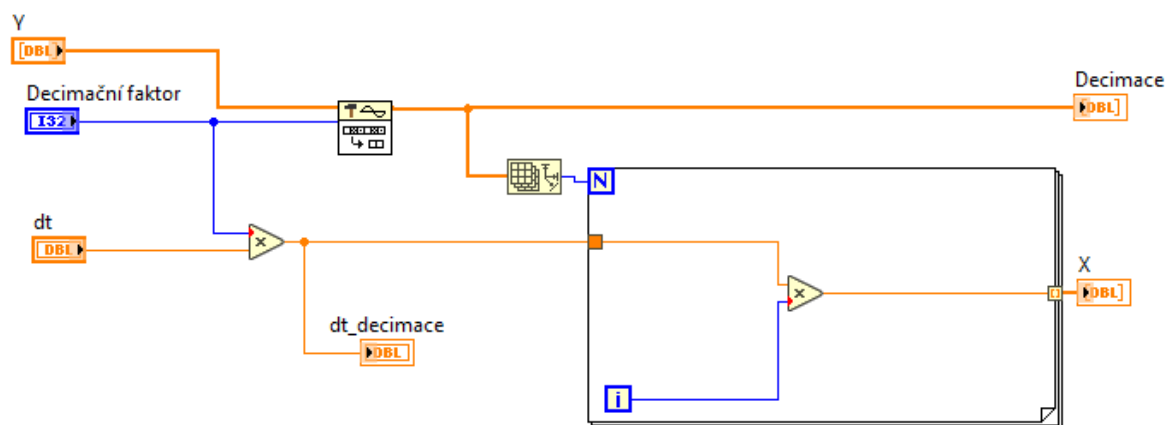


Obrázek 38: Graf úspěšnosti filtrace, Škoda Felicie.

Dále byl zkoumán vliv převzorkování na filtraci. Pro převzorkování signálu řeči a šumu byla použita funkce *Decimace_sub*, viz Obrázek 39. Tato funkce přepočítává přivedené pole hodnot podle zvoleného decimálního faktoru, viz Tabulka 9. Při decimaci je nutné zpětně rekonstruovat časovou osu, kdy dochází ke „smrštění“ signálu. To je ošetřeno tak, že se vynásobí hodnota kroku mezi vzorky původního signálu s decimálním faktorem. Hodnota nového kroku mezi vzorky je přivedena do smyčky *For*. Ta zajišťuje vykreslení správného času, tak že na vstup je přivedena hodnota velikosti pole pomocí funkce *array size*. Ve smyčce je následně vynásobena hodnota kroku mezi vzorky s iterací smyčky. Tak je zajištěno, že pro každý krok je přiřazen odpovídající čas.

Tabulka 9: Decimální faktor přepočítaný na frekvenci.

Decimální faktor (-)	1	2	3	4	5	6
Frekvence (Hz)	51200	25600	17066	12300	10240	8533



Obrázek 39: Decimace signálu.

Prvním krokem bylo nutné zjistit, jestli bude mít převzorkování vliv na hodnotu globálního SNR vstupních signálů. Ukázalo se, že rozdíly globálního SNR jsou minimální, jak ukazuje Tabulka 10.

Tabulka 10: Vliv převzorkování na hodnoty SNR pro vstupní signál.

Decimační faktor (-)	1	2	3	4	5	6
SNR (dB)	-2,07	-2,07	-2,05	-2,05	-2,07	-2,05

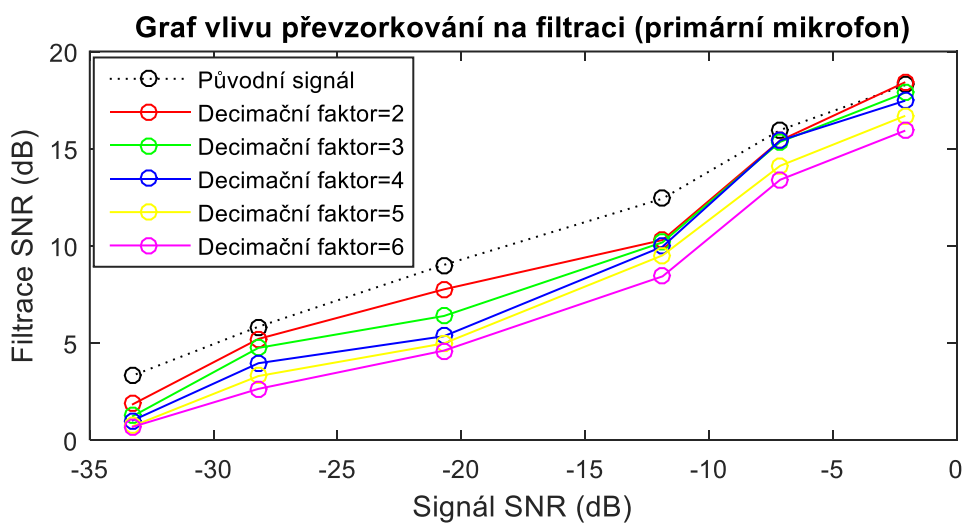
Z praxe se považuje za efektivní vzorkovat na co možná nejnížší přijatelné vzorkovací frekvenci z důvodu snížení matematické zátěže, tedy snížení požadavků na hardware a minimalizaci objemu dat. Z experimentů vyplývá, že tato myšlenka není vhodná pro adaptivní filtraci. Z experimentu totiž vyplývá, že při převzorkování na nižší hodnotu budou adaptivní filtry vykazovat horší filtraci. U algoritmu LMS se hodnota filtrace mezi původním signálem a nejvyšším decimačním krokem pohybuje s poklesem kolem -3 dB. Tento rozdíl už je lidské ucho schopné zaznamenat. Dále bylo u algoritmu LMS zjištěno, že čím bude výkon šumu oproti výkonu signálu větší, tím musí být hodnota konvergenční konstanty menší, aby se algoritmus dokázal optimálně adaptovat. Podobně to platí i pro nastavování délky filtru, kdy se ukázalo, že neoptimálnější hodnota je co nejmenší, tedy 2.

U algoritmu NLMS bychom mohli říct, že je převzorkování vhodné ale jen pro hodnoty výkonu šumu do zhruba -7 dB. Pak se výsledky globálního SNR podobají s jeho standardní variantou, algoritmem LMS

Nejhůře dopadl algoritmus RLS, kdy byl zaznamenán nejvyšší pokles až o -16 dB. Je to z důvodu, že algoritmus přepočítává i hodnoty z předešlé iterace, a tudíž algoritmus nemá dostatečný počet vzorků, aby mohl efektivně filtrovat. I přes tyto nevýhody ale dopadl v porovnání s ostatními algoritmy nejlépe. U algoritmu QR-RLS byly hodnoty filtrace a výpočetní doba totožné jako u algoritmu RLS, proto byly uvedeny pouze výsledky pro algoritmus RLS. Je to pravděpodobně z toho důvodu, že Adaptive Filter Toolkit v LabVIEW obsahuje dva totožné funkční bloky.

Tabulka 11: Výsledky filtrace pro algoritmus LMS, elektromobil.

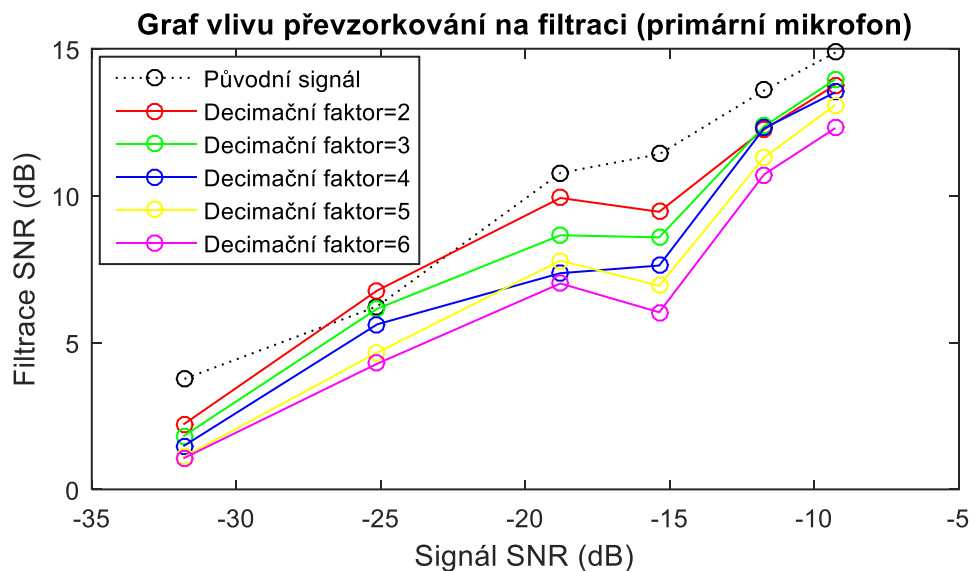
Typ šumu	SNR (dB)						
	Signál	LMS 1	LMS 2	LMS 3	LMS 4	LMS 5	LMS 6
Zavřená okna 50 km/h	-2,07	18,27	18,42	17,88	17,47	16,69	15,93
Zavřená okna 90 km/h	-7,12	15,96	15,42	15,35	15,42	14,12	13,40
Zavřená okna 130 km/h	-11,87	12,43	10,30	10,18	9,96	9,51	8,43
Otevřená okna 50 km/h	-20,70	9,02	7,76	6,39	5,35	4,98	4,61
Otevřená okna 90 km/h	-28,19	5,84	5,21	4,75	3,96	3,30	2,64
Otevřená okna 130 km/h	-33,26	3,33	1,85	1,24	1,02	0,73	0,69



Obrázek 40: Vliv převzorkování na filtraci pro algoritmus LMS, elektromobil.

Tabulka 12: Výsledky filtrace pro algoritmus LMS, Škoda Felicie.

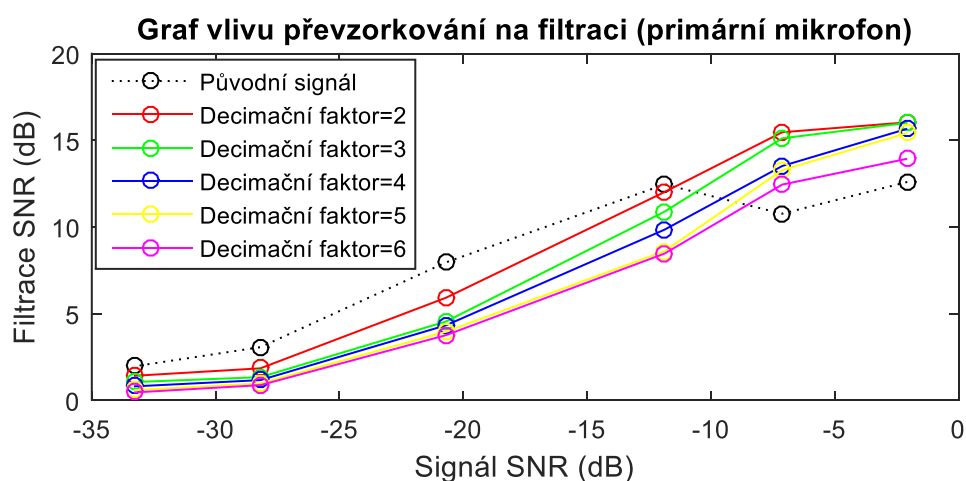
Typ šumu	SNR (dB)						
	Signál	LMS 1	LMS 2	LMS 3	LMS 4	LMS 5	LMS 6
Zavřená okna 50 km/h	-9,26	14,87	13,77	13,96	13,52	13,09	12,30
Zavřená okna 90 km/h	-11,73	13,58	12,25	12,36	12,29	11,30	10,72
Zavřená okna 130 km/h	-15,36	11,41	9,45	8,58	7,62	6,93	6,03
Otevřená okna 50 km/h	-18,78	10,77	9,92	8,65	7,36	7,79	7,01
Otevřená okna 90 km/h	-25,12	6,23	6,77	6,15	5,62	4,66	4,28
Otevřená okna 130 km/h	-31,80	3,76	2,23	1,83	1,49	1,11	1,07



Obrázek 41: Výsledky filtrace pro algoritmus LMS, Škoda Felicie.

Tabulka 13: Výsledky filtrace pro algoritmus NLMS, elektromobil.

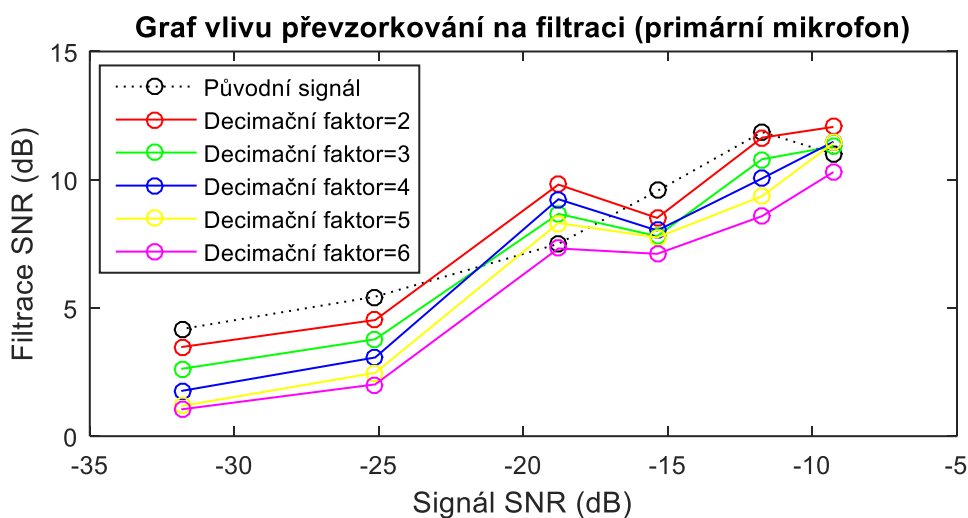
Typ šumu	SNR (dB)						
	Signál	NLMS 1	NLMS 2	NLMS 3	NLMS 4	NLMS 5	NLMS 6
Zavřená okna 50 km/h	-2,07	12,60	16,04	16,01	15,67	15,43	13,94
Zavřená okna 90 km/h	-7,12	10,77	15,47	15,12	13,52	13,28	12,45
Zavřená okna 130 km/h	-11,87	12,46	12,02	10,89	9,85	8,61	8,47
Otevřená okna 50 km/h	-20,70	7,96	5,92	4,55	4,33	3,96	3,76
Otevřená okna 90 km/h	-28,19	3,07	1,85	1,34	1,17	0,98	0,88
Otevřená okna 130 km/h	-33,26	1,97	1,42	1,06	0,81	0,59	0,46



Obrázek 42: Výsledky filtrace pro algoritmus NLMS, elektromobil.

Tabulka 14: Výsledky filtrace pro algoritmus NLMS, Škoda Felicie.

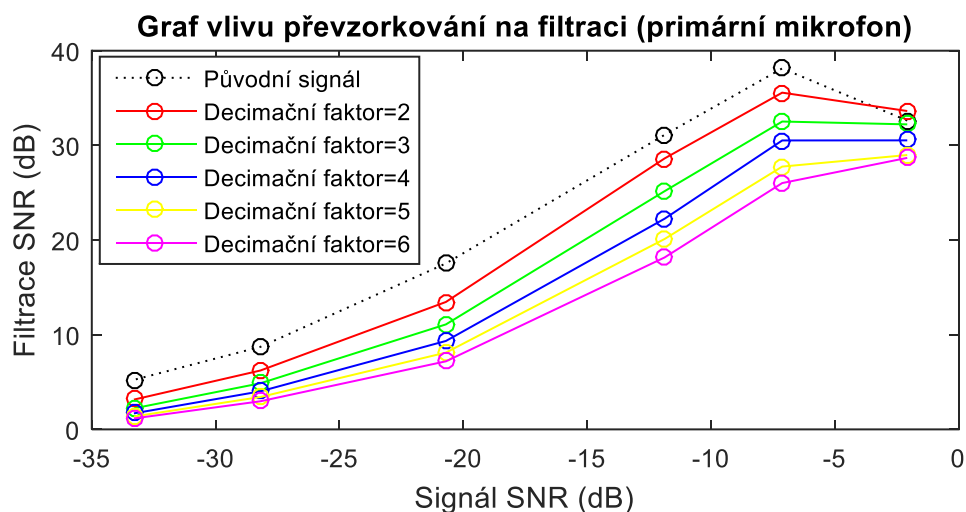
Typ šumu	SNR (dB)						
	Signál	NLMS 1	NLMS 2	NLMS 3	NLMS 4	NLMS 5	NLMS 6
Zavřená okna 50 km/h	-9,26	11,01	12,06	11,30	11,48	11,43	10,30
Zavřená okna 90 km/h	-11,73	11,86	11,64	10,80	10,05	9,36	8,58
Zavřená okna 130 km/h	-15,36	9,56	8,51	7,80	8,02	7,75	7,11
Otevřená okna 50 km/h	-18,78	7,50	9,81	8,67	9,24	8,31	7,32
Otevřená okna 90 km/h	-25,12	5,44	4,53	3,78	3,07	2,47	2,02
Otevřená okna 130 km/h	-31,80	4,18	3,48	2,63	1,77	1,18	1,05



Obrázek 43: Výsledky filtrace pro algoritmus NLMS, Škoda Felicie.

Tabulka 15: Výsledky filtrace pro algoritmus RLS, elektromobil.

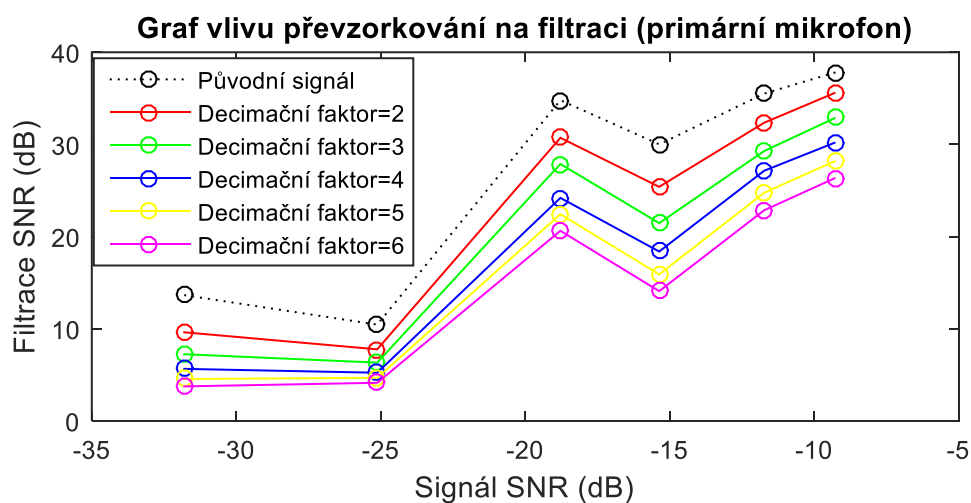
Typ šumu	SNR (dB)						
	Signál	RLS 1	RLS 2	RLS 3	RLS 4	RLS 5	RLS 6
Zavřená okna 50 km/h	-2,07	32,62	33,63	32,22	30,53	28,98	28,68
Zavřená okna 90 km/h	-7,12	38,13	35,57	32,52	30,52	27,76	26,02
Zavřená okna 130 km/h	-11,87	31,11	28,62	25,16	22,25	20,11	18,16
Otevřená okna 50 km/h	-20,70	17,53	13,46	11,09	9,34	8,11	7,19
Otevřená okna 90 km/h	-28,19	8,72	6,22	4,87	4,02	3,42	2,97
Otevřená okna 130 km/h	-33,26	5,26	3,17	2,24	1,71	1,38	1,17



Obrázek 44: Výsledky filtrace pro algoritmus RLS, elektromobil.

Tabulka 16: Výsledky filtrace pro algoritmus RLS, Škoda Felicie.

Typ šumu	SNR (dB)						
	Signál	RLS 1	RLS 2	RLS 3	RLS 4	RLS 5	RLS 6
Zavřená okna 50 km/h	-9,26	37,82	35,62	32,89	30,22	28,22	26,37
Zavřená okna 90 km/h	-11,73	35,51	32,35	29,28	27,14	24,75	22,85
Zavřená okna 130 km/h	-15,36	30,01	25,40	21,46	18,38	15,90	14,08
Otevřená okna 50 km/h	-18,78	34,80	30,72	27,88	24,20	22,43	20,63
Otevřená okna 90 km/h	-25,12	10,45	7,77	6,33	5,23	4,68	4,14
Otevřená okna 130 km/h	-31,80	13,65	9,62	7,24	5,65	4,55	3,75



Obrázek 45: Výsledky filtrace pro algoritmus RLS, Škoda Felicie.

Závěr

Cílem bakalářské práce bylo navrhnout a realizovat aplikaci v online i offline režimu ve vývojovém prostředí LabVIEW pro záznam a analýzu zvukových signálů v časové a frekvenční oblasti. Pomocí online aplikace byla vytvořena databáze řečových signálů a rušení vyskytujících se zejména v okolním prostředí. Tyto nahrávky byly následně použity v offline aplikaci v rámci prováděných experimentů, na které byly otestovány představitelné obou základních skupin adaptivních algoritmů LMS, RLS a jejich odvozeniny NLMS a QR-RLS.

Při zkoumání parametrů byla zjištěna u algoritmu LMS a NLMS menší rychlost konvergence a vyšší chyba filtračního procesu. Výhodou těchto algoritmů je jednoduchost a matematická nenáročnost.

Druhou skupinou byly algoritmy RLS a QR-RLS. Tyto algoritmy se vyznačují extrémně rychlou konvergencí a přesností na úkor složitosti a matematické náročnosti.

Pro testování účinnosti jednotlivých algoritmů byl proveden experiment, kdy bylo naměřeno rušení v kabině Škody Felicie a elektromobilu. Kritériem pro porovnání účinnosti jednotlivých algoritmů je především zvýšení poměru signálu od šumu, kdy byla zvolena metoda globálního SNR. Z experimentu bylo zjištěno, že nejlépe si vedl algoritmus RLS a QR-RLS, kdy hodnoty dosahovaly přes 30 dB. Oba algoritmy vykazovaly stejný výpočetní čas a hodnotu SNR. To je pravděpodobně způsobeno tím, že v knihovně Adaptive Filter Toolkit je funkce RLS zastoupena pro oba algoritmy.

V poslední části byl testován vliv převzorkování na filtraci. Pro vstupní převzorkované hodnoty vykazovalo globální SNR minimální rozdíly, avšak po odfiltrování signálu nastal znatelný pokles. Pro algoritmy LMS a NLMS se pokles pro maximální decimální krok pohyboval kolem -3 dB. Takový pokles už je dostatečně velký, aby jej dokázalo zachytit lidské ucho. U algoritmu RL a QR-RLS bylo zhoršení ještě výraznější, kdy hodnoty poklesu dosahovaly až -16 dB. To je způsobeno především tím, že algoritmy přepočítávají i hodnoty z minulé iterace, a tudíž nemají dostatečný počet vzorků na vysokou míru filtrace. I navzdory těmto velkým ztrátám byl však algoritmus RLS a QR-RLS úspěšnější.

Z praktického pohledu jsou požadovány co nejmenší náklady na realizaci adaptivních algoritmů a co největší míra filtrace. V tomto ohledu je mnohem výhodnější implementovat algoritmus LMS, který je schopen vysoké míry filtrace i přes jeho jednoduchost, avšak při vzrůstajícím výkonu a kvalitě výpočetní techniky se otevírá prostor i pro komplikovanější adaptivní algoritmy k řešení rozsáhlých úloh za relativně nízkou cenu.

Použitá literatura

- [1] PSUTKA, Josef. *Mluvíme s počítačem česky*. Praha: Academia, 2006. ISBN 80-200-1309-1.
- [2] MALENOVSKÝ, V. *Adaptivní filtrace zašuměných řečových signálů*. *Elektrorevue-Internetový časopis*. 2002, č. 63.
- [3] FARHANG-BOROUJENY, B. *Adaptive filters: theory and applications*. 2nd ed. Chichester: Wiley, 2013. ISBN 978-1-119-97954-8.
- [4] JAN J. *Číslicová filtrace, analýza a restaurace signálů*. 2. vydání Vutium Brno 2002, 427 s. ISBN 80-214-2911-9.
- [5] BELLANGER, Maurice. *Adaptive digital filters*. 2nd ed., rev. and expanded. New York: Marcel Dekker, c2001. ISBN 0-8247-0563-7.
- [6] SAEED V. VASEGHI. *Advanced Digital Signal Processing and Noise Reduction*. 3rd ed. Chichester: John Wiley Sons, 2006. ISBN 9780470094952.
- [7] MARTIENK, R. *Využití adaptivních algoritmů LMS a RLS v oblasti adaptivního potlačování šumu a rušení*. roč. 2013, č. 1, s. 8. URL:
< http://147.228.94.30/images/PDF/Rocnik2013/Cislo1_2013/r6c4c9.pdf > [cit. 2016 12-26].
- [8] NATIONAL INSTRUMENTS CORP. *Manuals–LabVIEW 2016 Adaptive Filter Toolkit Help*. National Instruments Corp, 2016.
- [9] UHLÍŘ, Jan. *Technologie hlasových komunikací*. Praha: Nakladatelství ČVUT, 2007. ISBN 978-80-01-03888-8.
- [10] POLLÁK, P.: Metody odhadu odstupu signálu od šumu v řečovém signálu, *Akustické listy*, vol. 7, no. 3, pp. 14-21, 2001
- [11] WITTASSEK, Tomáš. *Virtuální instrumentace I: učební text* [online]. 1. vyd. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, 2014 [cit. 2016-04-27]. ISBN 978-80-248-3395-8.

- [12] G. R. A. S. SOUND & VIBRATION. *http://www.gras.dk* [online]. [cit. 12.2.2017]. Dostupný na WWW: <http://www.gras.dk/40pp.html>
- [13] NATIONAL INSTRUMENT. *http://www.ni.com* [online]. [cit. 12.2.2017]. Dostupný na WWW: <http://www.ni.com/cs-cz/support/model.ni-9234.html>
- [14] KRAJNÍK, Eduard. *Základy maticového počtu*. Praha: Česká technika - nakladatelství ČVUT, 2006. ISBN 80-01-03376-7.
- [15] OPPENHEIM, Alan V. a Ronald W. SCHAFER. *Discrete-time signal processing*. Englewood Cliffs: Prentice-Hall International, c1989. ISBN 0-13-216771-9.

A. Příloha na DVD

Tabulka 17: Obsah přiloženého DVD.

Název adresáře	Popis
LabVIEW	Online a offline aplikace
Matlab	Implementace grafů
Nahrávky	Databáze nahrávek řeči a rušení
Text	Bakalářská práce v elektronické podobě a technický popis online aplikace